A Framework for Supporting
Anonymity
in Text-based Online Conversations

Andrew LEE Wei Tien

October 2001

A thesis submitted to Bond University in fulfilment of the requirements
for the Degree of Masters of Science in Computer Science

Abstract

This research has investigated how anonymity has been achieved in text-based online conversations. It has found that anonymity could be attained without any special provision from a conversation system. The absence of face-to-face contact and use of typed remarks are sufficient to create anonymity.

Nevertheless, the lack of special provisions can make it difficult for some to use the anonymity they have attained. Preserving such naturally attained anonymity can be equally difficult for users. System administrators will also have trouble controlling anonymity without special provisions. Will deliberate provisions for anonymity remove these problems?

The goal of this research is to determine how anonymity in online conversations could and should be supported. An existing conversation system lacking in special support for anonymity has been selected. Every possible change for the benefit of anonymity has been made to this system. The changes that have been made and why they were made are described in this thesis. The impact of those changes is also discussed.

The final outcome of this research is a set of guidelines and standards for supporting anonymity in text-based online conversations.

Table of Contents

Chapter 1

Introduction

Text-based online conversations and Anonymity

A *text-based online conversation* is a style of communication where two or more parties *exchange typed remarks over a computer network in real-time.*[1] Since there is no direct contact between the conversing parties, very little may be known or certain about the identity of one another.

A username (ie login name) may be all that one party knows about another—a point illustrated in the hypothetical (text-based) online conversation below:

```
bob: hey terminator
terminator: yes?
bob: what's your real name?
terminator: just call me terminator like everyone else
bob: but who are you?
terminator: i am terminator
```

Even if a person were to provide a 'real name', how would one know it was not simply something fabricated? In fact, everything the person said might have been fabricated. As the following hypothetical shows, someone using the name *Bill Gates* does not necessarily mean that the founder of *Microsoft Corporation* is online:

```
bob: who are you gates?
gates: Bill Gates
bob: i mean who are you in real-life?
gates: I am BILL GATES. I am Microsoft!
```

[1] The phrase *text-based online conversation* was an expression coined by the author because he believed it was more concise and accurate than terms such as: *synchronous computer-mediated communication* (Reid 1991), *real-time computer conferencing* (Ellis et el 1991), *text-based synchronous remote electronic meetings* (Rees et el 1993), *chat* (Shafer 1997) or *chatting* (Crumlish 1997), or *online synchronous conversations* (Suler 1997b).

```
bob: are you serious?
gates: Yes
```

Who are the people behind the names *Terminator* and *Bill Gates*?

An Anonymous user

The Oxford[2] dictionary explains that the adjective *anonymous* originates from the Greek word *anõnumos*, which literally means *nameless*. Understandably, the Oxford dictionary regards an 'anonymous' person as someone whose 'name is unknown.' Is the user 'terminator' anonymous? The word *name* needs to be clarified to answer this question.

Virtually all dictionaries (including the Oxford) define a *name* as a word by which a person is addressed (ie spoken of or to). In other words, *terminator* is a name. As such, the user 'terminator' is *not* anonymous by definition. However, the author argues that such an understanding of anonymity is not appropriate in an environment where everyone essentially has a 'name'—his or her username. He prefers to think of an anonymous person as *someone mysterious and not merely nameless*.

The name *terminator* does not make the user 'terminator' any less mysterious. Is 'terminator' a male or female? What does 'terminator' look like? How old is 'terminator'? Where does 'terminator' live? The author believes that the *real name* of 'terminator' (ie the name recorded in his or her birth certificate, passport, or identity card) is *at least* needed for these questions to be answered. The qualifier 'at least' is used since a person's real name may not even be sufficiently unique. In a world where there are billions of people, it is not unimaginable that there could be (several) people having the same full name. To simplify matters however, the author defines an anonymous person as *someone whose (full) real name is not known*. In other words, one's 'acquaintance' of ten years is still technically anonymous until one knows his or her full real name.

Some support for the author's 'definition' can be found in the Webster's[3] dictionary. It regards an anonymous person as someone 'with no name known or acknowledged.' In other words, someone who refuses to provide a name or has provided an unacknowledged (ie unrecognised) name is anonymous. That should be equivalent to saying that a person is anonymous unless his or her acknowledged name is known. Since a person's real name is perhaps the only name that is universally acknowledged, one should be able to assert that a person is anonymous unless his or her real name is known.

Is 'terminator' anonymous? A person's username can be (and is often) different from his or her real name. A full real name such as *Sarah Jane Parker* consists

---

2 *The Concise Oxford Dictionary of Current English (Seventh Edition).*

3 *Webster's New World College Dictionary (Third Edition).*

of three separate names: a first (given) name, followed by a middle (given) name, and a last name (ie family name).[4] In comparison, a username is usually a single word. It can be anything from the person's real name (eg *sarah_jane_parker or sarahjane_parker*) to a truncated version (eg *parker_s*, *sarah_j_parker*, *sarahjane*, *sarah*, or even *rah*), to something concocted (eg *sunrise* or *terminator*). It is clear that *terminator* is not someone's full real name. Since all that is known about 'terminator' is the name *terminator*, he or she must be anonymous.

Trying to deduce who a person is, based on a username, will be difficult if not impossible. A username such as *terminator* does not reveal anything about a user. Is the user a male or female? Even if someone were to have a more traditional username such as *sarah*, how would one know that 'sarah' is a female? How would one know that *Sarah* is the person's real name? Even if *Sarah* were the user's first name, the user's real name is not yet known. What is her last name? Without knowing her last name (and other given names), what she looks like, sounds like, or where she lives, it will be very difficult to determine which person is the user 'sarah'.

Is the user 'gates' anonymous? This user has revealed his identity—*Bill Gates*, the founder of *Microsoft Corporation*. If one accepts that 'gates' is the Bill Gates, then 'gates' is not anonymous. If one does not, 'gates' must be anonymous because 'he' (or 'she') has not admitted any other names.

An Anonymous remark

According to the Webster's dictionary, the adjective *anonymous* is also used to describe something 'given, written, etc. by a person whose name is withheld or unknown.' In other words, *anything done or made by someone that cannot be named is anonymous*. In the context of online conversations, an *anonymous remark* is a remark that is not tagged by a user's username (ie is a remark that cannot be traced to a user).

The remarks in the following hypothetical conversation are technically not anonymous because each is tagged by a username:

```
bob: excuse me people
bob: does anyone know whether i should buy Apple or Microsoft shares?
terminator: Microsoft of course!
gates: buy Apple shares while Steve Jobs is there
```

A name tag on one's jacket identifies one to strangers. In a similar way, a username tag on one's remarks identifies one's remarks. The tag on one's jacket can be removed. Similarly, the tags on one's remarks can also be removed. The username tags are present because of the underlying

[4] The composition of a person's full (real) name may differ according to culture and race. The author's own name is one such example where it begins with a given name, followed by a surname, and ends with two other given names.

conversation system.[5] Any conversation system can remove the username tags from the remarks of its users. Without the username tags, the remarks become anonymous:

```
excuse me people
does anyone know whether i should buy Apple or Microsoft shares?
Microsoft of course!
buy Apple shares while Steve Jobs is there
```

Is the remark below a non-anonymous remark? 'Yes,' by definition. However, there are two kinds of non-anonymous remarks.

```
terminator: Microsoft of course!
```

If one knew that 'terminator' was Sarah Parker (ie if 'terminator' were not anonymous), the remark above would in fact, be an *identified remark*. An identified remark is a non-anonymous remark. An identified remark can be traced to a specific person in the real world.

Some element of anonymity would exist in a remark if its author were anonymous. If 'terminator' were anonymous, the remark by 'terminator' cannot be traced to a person in the real world. However, the remark cannot be regarded as an anonymous remark because it can be traced to a user (ie 'terminator'). Nonetheless, the author sees no reason why a remark tagged by an anonymous username cannot be called a *semi-identified* (or *semi-anonymous*) *remark*.

Research issues

From the discussions in this chapter alone, three simple ways of achieving anonymity in online conversations (ie *conversational anonymity*) have been described:

1. by using a mysterious username
2. by pretending to be someone else (ie by using a false name)
3. by using a conversation system that does not tag remarks with usernames

Is conversational anonymity this easily attained in reality? How has conversational anonymity been supported? What problems may confront someone anonymous or someone seeking anonymity? *How can or should conversational anonymity be supported?* These are the research questions—the last being the research problem.

It is not the intention of this research to question the value or need for conversational anonymity. The use of anonymity as a personal protection

[5] A *text-based online conversation system* is a piece of software that enables people to engage in an online conversation. The phrase *conversation system* will often be used without the words *text-based* and *online* for brevity.

against such things as reprisal, prejudice, or ridicule is surely not new. Although anonymity is not always necessary or wanted, there is no doubt that it is useful to and sought by some people. In short, the author has accepted that conversational anonymity can be (or can be made) useful.

Chapters in this thesis

There are five other chapters in this thesis. Chapter 2 (*Preliminary Findings*) explains how conversation systems and services have been supporting conversational anonymity and why the author believed the support could be improved.

Chapter 3 (*In-depth Research Strategy*) describes specific research goals and the strategies developed to achieve them. Chapter 4 (*Implementation*) shows how the author's theories were implemented and refined.

Chapter 5 (*Analysis and Discussion*) evaluates the final outcomes. Chapter 6 (*Conclusion*) asserts the thesis of this research.

Chapter 2

Preliminary Findings

2.1 The existing path to Anonymity

The author has delved into the literature for answers to two important questions:

1 What provisions exist in text-based online conversation systems[6] and services for anonymity?

2 What tactics have users been using to attain conversational anonymity?

Continued literature searches could not provide the answers to these questions. Instead of hoping and waiting for someone to find the answers, the author decided to conduct a field study.

Various conversation services, systems, and source codes were examined. No claim is made that the field study was exhaustive. However, every conversation system or service that had openly claimed to support conversational anonymity was carefully examined. After two years of field study, the author found his answers.

2.1.1 Methods of attaining Anonymity

The **pretend (or deception) technique** is the use (or disclosure) of fabricated information. More specifically, it involves the adoption of a *false*

[6] Any piece of software that enables people to engage in a text-based online conversation can be considered a text-based online conversation system. Electronic mail (e-mail) systems however, do not qualify because the exchange of e-mail is an act of correspondence rather than conversation. The author defines a conversation as *an exchange of single or short remarks in real-time*. Systems that do not allow users to send and receive messages in real-time cannot be regarded as an online conversation system. Systems that support voice or video transmissions in addition to text messages (ie desktop or multimedia conferencing systems) also do not belong under *text-based online conversation systems*.

name (ie a name concocted or belonging to someone else) as one's 'full real name'. To make the deception more believable, various 'tricks' might be used—eg an e-mail account that reinforces one's false name,[7] a scanned photograph (of someone that matches one's concocted image), or a co-conspirator to back one's claims.

A person using the pretend technique will not seem mysterious (or anonymous for that matter). The person will not appear to be secretive about his or her identity. In reality, such a person is a mystery because much of what is known about the person is untrue and useless. Such a person is anonymous because his or her real name remains unknown.

A less devious approach can be called the **alias (or nondisclosure) technique**. This is where one *openly* refuses to reveal one's real name. One withholds information instead of giving false information. A name will be used (usually one's username) but it will be obvious to others that the name is not one's full real name. The name (or more exactly, alias) will appear to be too odd or whimsical (eg *Batman* or *Terminator*), improbable (eg *Elvis Presley* or *Marilyn Monroe*), or incomplete (eg *Bill* or *Sarah*). People will know that one is anonymous (or trying to be anonymous).

The pretend and alias techniques can be used on any conversation system. They do not require any special provision to be made by the underlying conversation system. The absence of face-to-face contact and use of typed messages are all that is required to support the pretend and alias techniques. These conditions are 'naturally' present when people converse online.

The third technique is only possible on certain conversation systems. The **nameless technique** creates anonymity by allowing users to make anonymous remarks. It usually involves one having to select an option that instructs the conversation system to remove one's username from one's remarks.

Two types of Anonymity

It should be slowly becoming obvious that the 'type' of anonymity created by the nameless technique is different to that created by the alias or pretend technique. The alias and pretend techniques make a user anonymous while the nameless technique makes the user's remarks anonymous.

The pretend and alias techniques create what the author calls **identity anonymity**. Identity anonymity exists when a user's full real name is not known. It exists when a username cannot be traced to a real name (or when a user cannot be traced to a physical person). A question such as 'Who is Terminator?' or 'Is Sarah Parker using the name *Terminator*?' indicates that identity anonymity exists.

---

[7] If one were to pretend to be 'Sarah Parker', one might have an e-mail address such as `sarah_parker@hotmail.com`

Authorship anonymity is created when a remark cannot be traced to a user. Authorship anonymity is recognised when a question such as 'What did Terminator say?', 'Who said this?', or 'What did Sarah Parker say?' is asked. The nameless technique creates authorship anonymity.

Identity anonymity and authorship anonymity can exist together or on their own. Both are present when a remark cannot be traced to someone in the real world. Identity anonymity but not authorship anonymity would exist if the remark were traced to an anonymous user. Authorship anonymity but not identity anonymity would exist if a group of identified users[8] were able to use the nameless technique to make anonymous remarks—the participants would not be anonymous but their remarks would.

2.1.2 Obstacles to Anonymity

It was not always possible to use a particular technique successfully. Hence, it is not always possible to attain a specific type or level of anonymity.

The *design of a conversation system* for example, determined whether the nameless technique could be used. Unlike the alias or pretend technique, the nameless technique can only be used if the underlying system permitted the removal of the username tags from remarks. One cannot force a system to remove one's username from one's remarks. Many systems were not designed to support 'nameless' remarks. Many did not provide any method of creating anonymous remarks. On such systems, authorship anonymity could not be attained.

Even when the nameless technique was supported, authorship anonymity might not always be attained. If there were only *two* participants in a meeting, a remark not made by one must obviously belong to the other participant!

Idiosyncrasies (ie one's peculiarities and eccentricities)[9] that exist in one's remarks can also foil the nameless technique. If 'bob' knew that 'terminator' always typed in upper case, 'bob' would still be able to identify the remarks made by 'terminator' (even when the username *terminator* was not present):

```
bob: does anyone know whether i should buy Apple or Microsoft shares?
MICROSOFT OF COURSE!
```

Identity anonymity cannot always be attained either. *Circumstances* may not permit one to use a mysterious username. Discussions of a confidential or private nature for example, may only be open to identified users. If one were

[8] ie an *identified user* is a user whose real name is known.

[9] For example, spellings or misspellings of certain words, grammatical errors, or use of certain *smileys* or phrases. A smiley is simply an arrangement of characters that attempts to show one's facial expression—eg `:-)` or `:)` to represent a smile on one's face.

forced to use an *identified username*,[10] identity anonymity cannot be attained.[11]

Identity anonymity may also be prevented if one's *network address were not protected*. Every computer connected to a network should have a unique network address. A computer connected to the Internet for example, will have a unique *Internet Protocol address* (IP address). A conversation system essentially knows the network addresses of all its users (because the addresses are used to route messages to each user's computer). The fact that the system knows the addresses is not the problem. The problem occurs when the system exposes one's network address to the system administrator or other users. It may be possible to trace a network address to a particular computer (or locale) and ultimately, to a person. When network addresses are exposed, one may have to connect from a multi-user host computer or a computer shared by several people (such as one in a computer laboratory or an Internet cafe)[12] to attain and retain identity anonymity.

2.1.3 Problems after attaining Anonymity

Authorship anonymity (or more precisely the nameless technique) can *hinder communication*. Examine the following scenario.

'Bob' questions a group of people anonymously:

```
should I buy Apple or Microsoft shares?
```

Since 'terminator' does not know who asked the question, *the reply cannot be made in private.*[13] The user 'Terminator' is forced to respond in public:

```
terminator: Microsoft of course!
```

'Gates' can only respond in public as well. However, 'he' decides to do it anonymously:

```
buy Apple shares while Steve Jobs is there
```

Since 'bob' does not know that 'gates' suggested Apple shares, 'bob' cannot question 'gates' in private. If every remark were tagged by the appropriate

[10] An identified username is the username of an identified user. In other words, an identified username can be traced to a real name.

[11] Of course, one could always resort to the nameless technique (ie authorship anonymity) if it were supported. The fact that one's username is not anonymous will not matter because one's remarks would not be tagged by one's (non-anonymous) username.

[12] An Internet cafe (or Net cafe) provides public Internet-access in addition to the usual services of a café.

[13] Before a private message can be delivered, a recipient (or more precisely, the recipient's username) usually needs to be specified.

username (ie if the nameless technique were not used), no one would have any problem making private remarks.

The alias and pretend techniques can also create predicaments. Once a person becomes anonymous (ie attains identity anonymity), the person *cannot make identified remarks* (ie remarks that the person can take credit). Identified remarks require the use of an identified username. In other words, *one cannot be anonymous and yet make identified remarks*. To be able to make identified remarks, 'terminator' has to reveal and prove that she is Sarah Parker.[14] Only then will people know that a remark belonging to 'terminator' is a remark belonging to Sarah Parker.

If one wanted to make both identified and semi-identified remarks, *two* usernames would be needed—one identified and one anonymous.[15] Sarah will need to be 'terminator' when she wants to make identified remarks. She will need to use an anonymous username such as *mickey* when she does not want to make identified remarks. A change of username usually requires a user to leave a conversation (ie meeting), exit the system, re-enter the system under a different username, and return to the meeting. At the very least, this is *troublesome*.

Change of usernames may *not go undetected* either. Returning too soon may expose that the user that left and the one that arrived are the same person! If the return were delayed, the topic of conversation could have changed or worse, the meeting could have ended! Exposed network addresses can also foil username changes. If the network addresses of two users were identical, one could speculate that the two are the same person. In fact, it should be possible to tell that two users are the same person if both have the same idiosyncrasies.

An unsuccessful changeover if not realised by a user, can be a serious problem. There is *incalculable danger believing one is anonymous when one is not in reality*. In theory, the amount of danger would be proportional to the amount of protection anonymity has provided. There is little danger when one has used anonymity to make an inconsequential remark. On the other hand, there may be dire consequences if one had made complaints against one's employer.

---

[14] The author would like to remind the reader that 'Sarah Parker' is not a real person.

[15] The author would like to remind the reader that *a semi-identified* (or *semi-anonymous*) *remark* is different to *an anonymous remark*. A semi-identified remark is a remark that points to an unidentified (ie anonymous) user or username. In comparison, an anonymous remark does not point to any user. Semi-identified remarks are the result of identity anonymity while anonymous remarks create (or are the result of) authorship anonymity.

Loss of Anonymity

Although highly unlikely, 'terminator' may accidentally reveal that 'he' is Sarah Parker. A slip-up may also expose the fact that one is not who one claims to be. On one occasion, the author had accidentally used the pronoun *his* when he should have used *her*. This single blunder foiled weeks of successful pretence. The possibility of making a slip-up is ever present.

Identity anonymity can usually be restored by changing one's username. The problem is that one will become a stranger. Certain meetings may not be open to a stranger. Certain people may not want to speak to a stranger. All the *merits, reputation, and relationships* gained through the exposed username would be lost (or would have to be forgone).

Of course, loss of identity anonymity can also happen without one's knowledge. Much can be discovered about a person by *eavesdropping* on the person's conversations. Any conversation system can be built with eavesdropping capabilities. In fact, some systems deliberately support this capability! Sarah may be 'terminator' to one user and 'Peter Smith' to another. Sarah may be herself (ie Sarah Parker) to yet another. By eavesdropping on the conversations of 'terminator', one may learn that the user 'terminator' is Sarah Parker in real-life. Intrusions on privacy may never be known to anyone except the intruder.

Loss of authorship anonymity can also occur without one's knowledge. Again, any conversation system can be built to give certain people (eg a system administrator) the ability to deanonymise anonymous remarks. There is also incalculable danger in thinking that one's remarks are anonymous when they are not.

Authorship anonymity may also be lost because of prolonged interaction. The longer a group of people interact, the more obvious each person's idiosyncrasies become. Patterns not initially obvious may begin to emerge. Even if a remark cannot be traced to a specific user, it may now be possible to *determine which remarks have come from the same source*. When that happens, some authorship anonymity has been lost. When that happens, certain remarks become semi-anonymous (or semi-identified) instead of truly anonymous.

Once one has put anonymity to real use, one has to begin worrying about being exposed. The potential for loss of anonymity does not end just because one has stopped embracing anonymity (ie stopped using an anonymous alias). The danger of being exposed only ends when people have stopped trying to deanonymise one's (anonymous) remarks or usernames. Such may be the price of anonymity.

Relying on multiple techniques should better protect a user. Sarah Parker for example, could assume the name *Peter Smith* (ie the pretend technique) and use the alias *terminator* as her username (ie the alias technique). When she has to make 'risky' remarks, she could also resort to the nameless technique (in addition to the tag and pretend techniques). Should the nameless technique fail, her remarks would only be traced to the user 'terminator'. Sarah would

still have the protection of (identity) anonymity. Should the alias technique fail, Sarah would still enjoy identity anonymity because the user 'terminator' would point to *Peter Smith* (which does not exist in reality). Sarah will continue to enjoy some form or level of anonymity until people are able to trace her remarks to her full real name, *Sarah Parker*.

2.2 Case studies

The following is a selection of systems and services that illustrate and elaborate the points made in the previous section.

2.2.1 The UNIX Talk program

The *Talk* program is designed to enable two UNIX users to engage in an online conversation from within their shell accounts. Talk works like a telephone conversation. One party will initiate the 'call'. Instead of a telephone number, the account detail of the 'receiver' (ie a combination of the receiver's login name and IP address) is provided. If the receiver accepts the call, the screens of both parties are divided into two equal portions. Whatever one party types is displayed in one of the portions on both screens. Transcript 2.1 shows how the screen of 'sarah' may look after a brief conversation.[16] The text in **bold** represents the text 'sarah' had typed.

```
hi sarah
this is a surprise. do i know you?
yes
from the beginning?
it may take a while




------------------------------------------------------------------------
Hello Bob
No you don't but it doesn't matter that you don't.
Have you seen Star Wars Episode 1?
Well, please tell me about it :)
Yes, if you don't mind.
I'm all ears! :)
```

Transcript 2.1: One's remarks always appear in the lower portion

[16] As with all the transcripts throughout this chapter, the people are fictitious and the conversations have been mocked. Nonetheless, the transcripts themselves are real because they were generated using actual systems.

Talk has no support for anonymity. In fact, it should not be used if one requires anonymity. Although the remarks one makes are not tagged by one's username, authorship anonymity cannot be attained because there are only two parties in the conversation. At least *three* people must be in a conversation before the nameless technique can generate authorship anonymity.

Identity anonymity is possible but with some difficulty. Talk requires that each party know the shell account of the other. In a telephone conversation, the telephone number of the caller can be hidden. In a Talk conversation, the 'telephone numbers' (ie the shell accounts) of both parties are exposed.

Since 'bob' knows that 'sarah' is `sarah@surf.bond.edu.au`, 'bob' can use the `finger` command in UNIX to learn more about 'sarah':

```
finger sarah@surf.bond.edu.au
Login name: sarah                       In real life: Sarah Parker
Directory: /home/sarah                  Shell: /bin/csh
On since Feb 15 00:03:53 on ttyqb from surf
No Plan.
```

If Sarah wanted to be anonymous, details of her real name would have to be removed:

```
finger sarah@surf.bond.edu.au
Login name: sarah                       In real life: Not Available
Directory: /home/sarah                  Shell: /bin/csh
On since Feb 15 00:03:53 on ttyqb from surf
No Plan.
```

Her username (ie login name) would also need to be more anonymous. She should have used a username like 'terminator':

```
finger terminator@surf.bond.edu.au
Login name: terminator                  In real life: Not Available
Directory: /home/terminator             Shell: /bin/csh
On since Feb 15 00:03:53 on ttyqb from surf
No Plan.
```

Alternatively, she could have impersonated the *Duchess of York*:

```
finger sarah@surf.bond.edu.au
Login name: sarah                       In real life: Sarah Ferguson
Directory: /home/sarah                  Shell: /bin/csh
On since Feb 15 00:03:53 on ttyqb from surf
No Plan.
```

Of course, Sarah could have simply concocted a fictitious identity:

```
finger jane_smith@surf.bond.edu.au
Login name: jane_smith                  In real life: Jane Smith
Directory: /home/jane_smith             Shell: /bin/csh
On since Feb 15 00:03:53 on ttyqb from surf
No Plan.
```

Such changes to one's shell account are only possible by the root user (ie the system administrator of *surf*). Unless 'sarah' can make (or get someone to make) any of those changes she may not be able to attain identity anonymity. Even then, everything may be in vain if the system administrator were someone that did not respect *confidentiality* (ie were someone that would tell 'bob' that 'sarah', 'terminator', or 'jane_smith' is *Sarah Parker* in real-life). Before Sarah can be anonymous to anyone, she may need to be anonymous to her system administrator or Internet Service Provider. She may have to utilise the pretend technique from the onset (ie have the system administrator believe that she is 'Jane Smith').

2.2.2 Internet Relay Chat

Whether people required anonymity or not, the *Internet Relay Chat* (IRC) was an improvement over the Talk program. IRC allows people to choose a username that is different to their shell account.[17] Furthermore, conversations are not limited to two people.

Although multi-party conversation is supported, authorship anonymity remains impossible. That is because IRC does not support the creation of nameless remarks. Every remark is tagged by a username—see Transcript 2.2. The text in **bold** represents what the user 'bob' had typed.

```
*** terminator has joined channel #fishing
<terminator> Hello everyone!
hello terminator
hello terminator
<terminator> What are you two up to?
<sam> planning a fishing trip
/whois terminator
*** terminator is ~terminator@modem1.bond.edu.au
*** on channels: #fishing
*** on irc via server irc-2.mit.edu :Massachusetts Institute of Technology
*** terminator has been idle 2 seconds.
<terminator> Can I come?
<sam> sure :)
i'm afraid terminator can't
i'm afraid terminator can't
<terminator> Why not? :(
<sam> why not bob?
err, because he's in Australia and we're in Michigan?
err, because he's in Australia and we're in Michigan?
```

Transcript 2.2: Every remark will be tagged by a username in IRC (bob's view)

The `/who` command allows users to determine the IP address of each other. This provides a way to authenticate one another. Of course, it also makes

[17] In fact, a UNIX shell account is not needed to use IRC.

attaining identity anonymity more difficult than it should. To prevent people from tracing one's IP address to one's 'door steps', one's personal computer should not be used. Depending on the amount of anonymity required (or the consequences of losing anonymity), one may have to resort to a computer open to the public—such as a computer in an Internet cafe or a coin-operated Internet booth in a shopping mall.

As a consolation, the username and IP address combination (ie `terminator@modem1.bond.edu.au`) may not necessarily be the e-mail address or shell account of 'terminator'. Since IRC does not require any party to have a shell account, performing a *finger* on `terminator@modem1.bond.edu.au` may not return any useful or the right information.[18]

Nevertheless, two things will be known about 'terminator'. 'Bob' will know that 'terminator' is connected from Australia (because of the `.au`) and is perhaps a student or staff at *Bond University* (because looking up `www.bond.edu.au` would have brought 'bob' to the university's web page). Although 'terminator' (ie Sarah Parker) is still anonymous (to 'bob'), she would have been 'more' anonymous if her IP address were never exposed.

Does IRC have any provision that is useful to anonymity? The author has identified two potentially useful provisions. First, IRC does not allow users to reserve usernames. In other words, the 'terminator' one day may not be the same 'terminator' the next. By using a very common username (eg 'john', 'jane', 'bob', 'mary'...), one would essentially be sharing a username with other people. As a result, one's idiosyncrasies become harder to isolate because it would be fused with those of others. Unfortunately, the fact that IP addresses are exposed means that it is possible for people to distinguish one 'john' from another (since the username and IP address combination will be reasonably unique).

The other potentially useful feature is the `/nick` command. It allows a user to change his or her username without having to leave the system—see Transcript 2.3 below. Such a provision could make the *transition to and from anonymity less troublesome*. However, it is unfortunate that IRC announces changes to one's username. As such, the `/nick` command was not as helpful to anonymity as it could have been.

```
/nick shark
*** bob is now known as shark
i like shark better
i like shark better
```

[18] IRC clients exist for all major operating systems.

```
*** terminator is now known as termite
that's cute
that's cute
<termite> Yes I do think so too.
```

Transcript 2.3: Changing nicks[19] using the **/nick** command (bob's view)

2.2.3 Town Meeting

Town Meeting 2.0 (TM) by *Adam Stein* was the *only* conversation system found to have openly claimed support for conversational anonymity: 'Anonymous mode available—everyone says what they really think.'[20]

TM supports identity anonymity in two ways: by allowing users to dictate their username, and by concealing their network addresses. Both provisions are extremely important as the TM client and server software runs on an *AppleTalk* local area network (instead of the Internet). If TM did not conceal the network addresses, one might be able run down the hallway to check who was at a particular computer!

Authorship anonymity is supported because TM supports the nameless technique. An option under one of the menus (see Figure 2.1) allows a user to switch between anonymous and non-anonymous[21] remarks. When the anonymous option is selected, the remarks that one creates appear without one's username.

Figure 2.1: Using the menus to toggle the Anonymous mode

Like all systems that have attempted to support the nameless technique, TM does not address the *two-person* problem. Assume that 'bill', 'ally', and 'sue' are the only people in a meeting. Someone has been making nameless (ie

[19] A *username* is called a *nick* (or *nickname*) in IRC.

[20] An excerpt from the *Town Meeting Read Me* file accompanying the software.

[21] The author categorises identified remarks (ie remarks traceable to an identified username) and semi-identified remarks (ie remarks traceable to an anonymous username) as non-anonymous remarks.

anonymous) remarks. Once 'ally' left the meeting however, 'bill' would have been able to trace the last two nameless remarks to 'sue' (since no other user was present)—see Transcript 2.4.

what do you think about the latest G3 Macs?
bill: what do you think about the latest G3 Macs?
they are getting cheaper
yes they are
bill: yes they are
think i should get one?
bill: think i should get one?
i don't like the way they look
the old G3 looked much better
ally has left the conference.

*At this point, 'bill' selects the **List People in Conference** command from the menus, causing TM to display a list of participants...*

People in Conference:
bill
sue
i agree. i don't like the new look either
bill: i agree. i don't like the new look either
can't i buy the older model and upgrade the processor?
bill: can't i buy the older model and upgrade the processor?
you can't
why not?
bill: why not?
the processor is stuck to the motherboard :-(

Transcript 2.4: The last two nameless remarks must belong to 'sue' (bill's view)

The last two 'anonymous' remarks would have been *semi-anonymous* or even *identified* (if the username *sue* were not anonymous). TM should have prevented 'sue' from creating a nameless remark because there were only two parties present.

It is possible for the anonymous mode to be selected before a user enters a conversation. If 'sue' had done that, her username would be replaced by *Anonymous*—an automatically generated alias.[22] When she enters a meeting, TM will announce that 'Anonymous has entered the conference' instead of 'sue has entered the conference.' The participants list will also show a participant called *Anonymous* instead of *sue*. If that were the case, 'bill' would have only known that the last two remarks were made by 'Anonymous'. 'He' should not know that 'sue' was present.

Do automatically generated usernames provide a way of addressing the two-person problem? The author believes they do but it will mean that the

[22] *Anonymous 2, Anonymous 3, Anonymous 4...* would be used if more than one person had preselected the anonymous mode.

anonymous mode cannot be disengaged for the duration of the meeting (or until other anonymous parties join the conversation). If 'sue' (ie 'Anonymous') were to deselect the anonymous mode, the 'sudden' introduction of the username *sue* should enable 'bill' to deduce that 'sue' was 'Anonymous'—thus, deanonymising any nameless remark previously traced to 'Anonymous'. *The author believes that preventing a user from making nameless remarks when there are fewer than three participants is a better solution.*

The TM implementation of the nameless technique also failed to address the most fundamental problem with authorship anonymity—*how do people respond to an anonymous remark privately?* TM allows private remarks to be directed to any automatically generated alias. However, how would one know which automatically generated alias should be the recipient (since the original remark would not be tagged by an automatically generated alias)? If 'Anonymous', 'Anonymous 2', and 'bill' were in a meeting, which of the two anonymous users should 'bill' direct the private response to?

Even if there were only one automatically generated alias (ie even if 'Anonymous', 'ally', and bill' were the only ones in a meeting), how could 'bill' be sure that a nameless remark did not come from 'ally'? Although 'ally' did not preselect the anonymous mode, 'she' would still be able to make anonymous remarks. Since 'bill' would not know whether 'ally' or 'Anonymous' made the nameless remark, he would have to make a public response—ie direct his response to both 'ally' and 'Anonymous'.

The author has also discovered a loophole (or conceptual flaw) that may enable someone to unravel the user hiding behind an automatically generated alias (ie determine that 'Anonymous' is actually 'sue'). Figure 2.2 to Figure 2.5 explains.

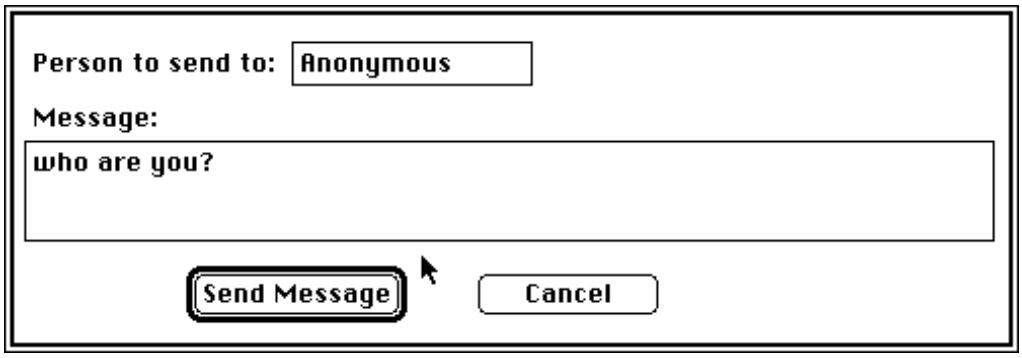

Figure 2.2: 'Bill' selects Send Message from the menus and begins composing a private message to 'Anonymous'[23]

[23] 'Bill' does not know that 'Anonymous' and 'sue' are the same person.

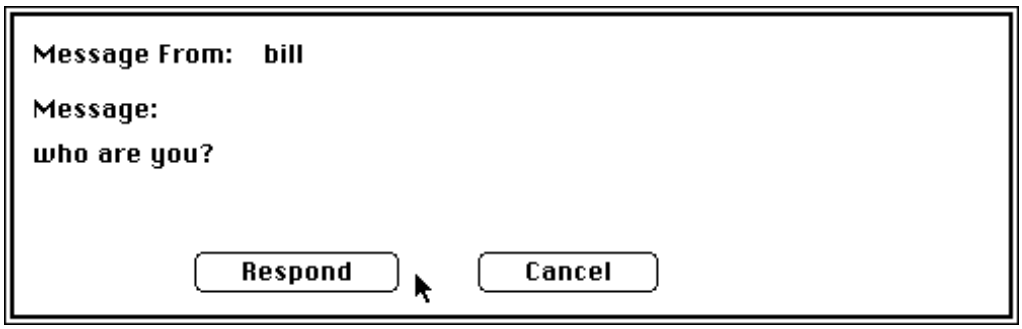

Figure 2.3: 'Anonymous' (ie 'sue') clicks on the **Respond** button to reply

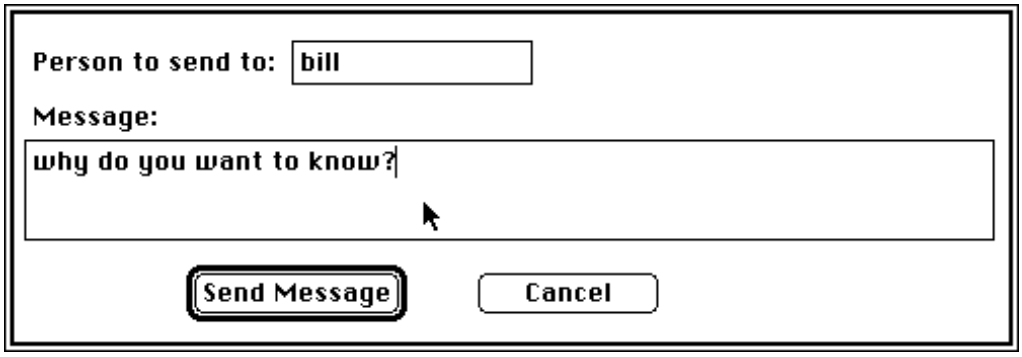

Figure 2.4: 'Anonymous' sends the reply

'Sue' should be identified as *Anonymous* because she has preselected the anonymous mode and has not had it deactivated. However, that is not what TM does...

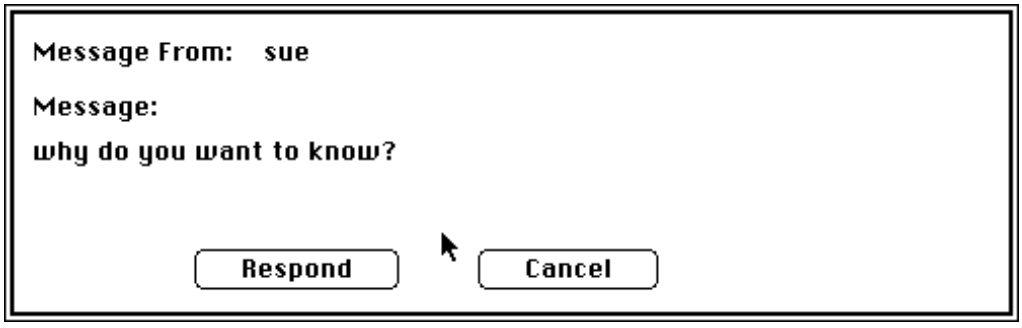

Figure 2.5: From the reply, 'bill' will know that 'Anonymous' is 'sue'

Eavesdropping may also be a concern on TM. Although TM does not specifically support eavesdropping, it can still occur. The server logs every remark except for private remarks created using the **Send Message** provision. The logs are displayed on the screen at the server. Two people might not use **Send Message** to converse if they were the only people in a meeting. While a third user might not be present in the system, there could be a *third person following the conversation at the server* (ie outside the system)! The system should have warned every user that every remark made without using **Send Message** would be recorded. Such a warning should be present each time a user joins a meeting or starts the software.

2.2.4 The Virtual-Eye System

Figure 2.6 shows two users having an online conversation in a 'multi-user virtual environment' called *Habitat* (Morningstar & Farmer 1990).

Figure 2.6: Two avatars engaging in an online conversation[24]

The two avatars[25] (ie users) appear to be able to recognise one another even though the avatars and remarks do not seem to be tagged by a username. The head and body of the avatars are different but that is certainly not enough to make every avatar unique (or recognisable). There must have been a way to tell the name of an avatar. What if there was no way of telling the avatars apart? Midway through the field study the author began creating what became the **Virtual-Eye** (ViE) system.

Usernames are not used in the ViE system. There are no participant lists. Instead of usernames, nondescript avatars are used. Every avatar is identical. They have no faces, features, or labels—see Figure 2.7 on the next page. Even if the users were to know one another in real-life, they should not be able to identify who was behind a particular avatar just by looking at an avatar because every avatar is identical.

[24] Source: `http://www.communities.com/picture/habitat.gif`

[25] An avatar is essentially an object on the screen representing a user (ie an icon of a user).

Figure 2.7: Three nondescript avatars

Meetings are held within virtual rooms. A virtual room is presented in a *first-person perspective* (ie a user would not see his or her avatar in the view). When a user makes a remark, a dialogue balloon will appear above the avatar that represents the user—see Figure 2.8.

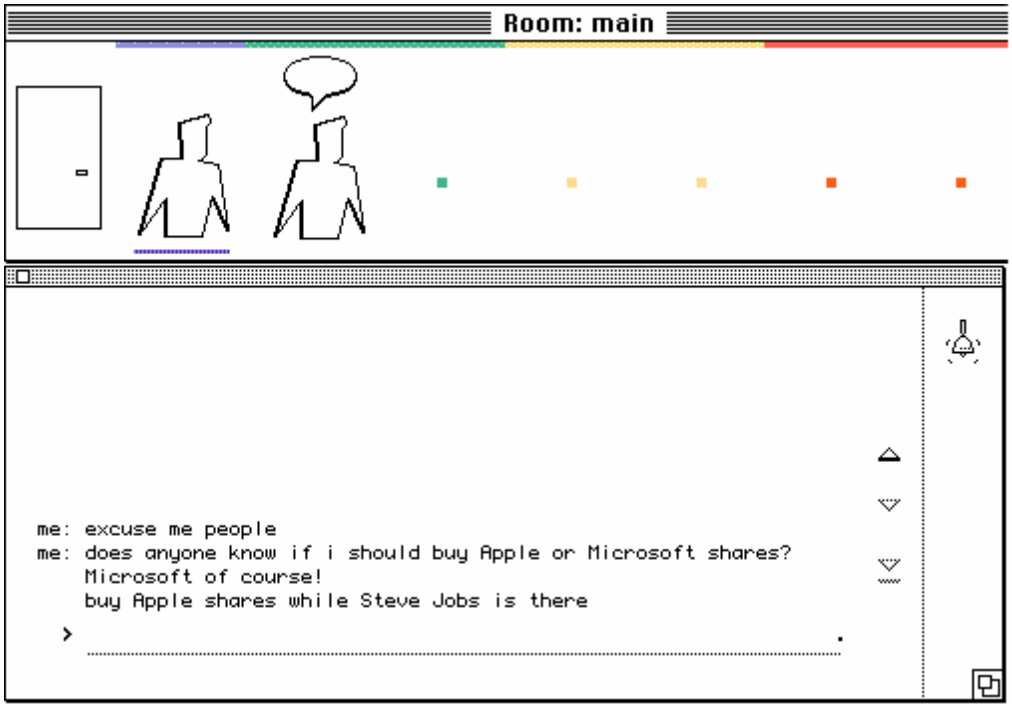

Figure 2.8: The dialogue balloon identifies the avatar that made the last remark[26]

The user's remark will not be tagged by any username. The only 'link' between a remark and an avatar will be the dialogue balloon. However, such a link is temporary. The balloon repositions itself as different people speak. Once the balloon moves away, the link between a remark and an avatar will only exist in a person's memory. If one were to forget, there is essentially no simple way of determining which remark belonged to which avatar (and vice-versa).

What is the purpose of the nondescript avatars and the dialogue balloon? When a user enters an occupied room for the first time, he or she would not see a bunch of avatars spread across the room. The avatars of the other

[26] The tag `me:` only appears on the screen of the speaker—helping the speaker to identify what he or she has said.

participants will be *grouped* as a group-avatar at the entry (ie left-most) position. This leaves seven free positions.[27] By dragging the balloon while it is above a group-avatar, the speaker can be separated—see Figure 2.9.[28]

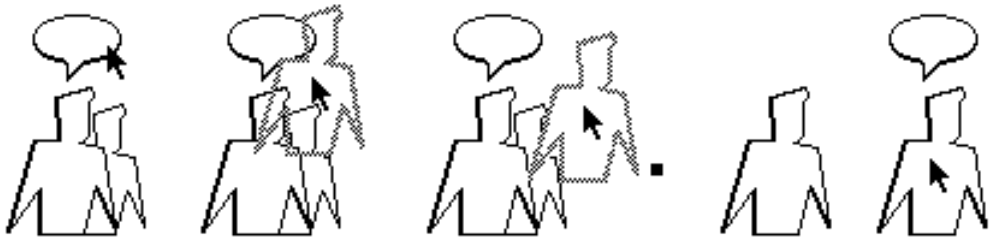

Figure 2.9: Separating the speaker from a group-avatar

Once separated a user can then 'double click' the avatar of the speaker to establish a private communication channel—see Figure 2.10. In essence, one can now finally question the speaker of an anonymous remark in private!

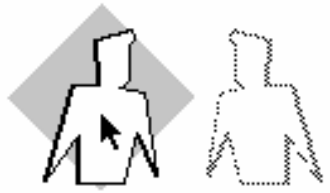

Figure 2.10: A diamond indicates that a private communication channel has been established with an avatar[29]

Individual avatars can be 'dragged' and 'dropped' onto each other to form new group-avatars—see Figure 2.11 below. By moving, grouping, and separating avatars, users can rank participants according to a criteria. An avatar (ie a person) with interesting ideas for example, can be separated and left to exist as an individual avatar. The avatar with the most interesting idea could be place further to the right. Those that have not made any useful contribution could be left as a group at the entry position. People of similar ideas could be grouped together so one could 'double click' the group-avatar to establish a private channel with a group of people.

Figure 2.11: Merging two individual avatars into a group

[27] A room has a maximum of eight positions on to which to place avatars (much like a row on a chess board).

[28] The reason for such a technique was to encourage selection by merit (by what a person says) rather than other attributes (eg the choice of a person's username).

[29] Once private communication is established, other avatars are greyed because one will no longer receive remarks from other avatars (ie from people represented by the other avatars).

Although the ViE system appeared to work in theory, it was less than ideal in practice. First, there is no sensible way for users to be non-anonymous or make non-anonymous remarks. That is because usernames are not supported. A user can add a name to his or her remark—eg 'buy Apple shares while Steve Jobs is there (Sarah Parker)'. The problem is that anyone can do the same—ie add '(Sarah Parker)' or any other name to their remarks. Too much emphasis has been placed on anonymity that non-anonymity was forgotten!

A greater problem however, is the fact that the conversation process has become very complicated. As soon as a meeting involves three or more parties, the conversation process is added with the need to:

- follow the movements of the dialogue balloon,
- rearrange avatars,
- remember the significance of avatars, and
- recall the significance of avatars.

The author halted all work on the ViE system because of these problems.[30]

2.2.5 The Mudde Pathetique MUD

Crumlish (1997) wrote 'I'm not going to go into the multi-user domains, or MUDs, that thrive on the Internet, powered mainly by the energy of college students staying up or procrastinating' under a chapter entitled *Chatting, Conferencing, and Virtual Worlds*.[31] While that may be true, the author strongly believes that to neglect MUDs is to create an imperfect perception of 'chatting' (ie online conversations). It might have been excusable to ignore MUDs when there were only 275 on the Internet.[32] In less than a year however, MUDs increased to 440.[33] As recently as April of 1999, the *MUD Connector*[34] listed 1,334 MUDs. The presence of MUDs on the Internet should not be ignored. MUDs are 'clever' conversation systems capable of supporting an almost face-to-face meeting experience.

---

[30] This thesis and Lee 1993 may well be the only record that the ViE system ever existed.

[31] Apart from *Multi-User Domain*, the acronym *MUD* has also been expanded to *Multiple User Dimension*, *Multiple User Dungeon*, or *Multiple User Dialogue* (Smith 1992)—all of which are synonymous.

[32] *Scott Geohring's The Totally Unofficial List of Internet Muds* (26 March 1993) published through the `rec.games.mud.announce` Internet newsgroup.

[33] *The Totally Unofficial List of Internet Mud* (22 January 1994) published through the `rec.games.mud.announce` Internet newsgroup. *This was the last list Geohring compiled.*

[34] `http://www.mudconnect.com/`

The Mudde Pathetique (MP)[35] is a MUD on the Internet. Transcript 2.5 attempts to show the type of interaction possible on MP.

```
Terminator has arrived.
look
Penny Lane
You are on Penny Lane.  Emerald Avenue is to the west and Penny Lane continues in
eastward direction.
Sam the Salmon King is standing here.
Terminator the good guy is standing here.
Obvious exits:
East  - Penny Lane
West  - Emerald Avenue
who
Players
-------
[ 1 Wa] Terminator the good guy
[ 1 Wa] Ally Cat
[ 1 Wa] Sue the Cowgirl
[ 1 Wa] Bob is JR's brother!
[ 3 MA] Sam the Salmon King
Terminator says "hello everyone!"
Terminator smiles happily.
smile
You smile happily.
say hello terminator
You say, "hello terminator"
Terminator says, "what are you two up to?"
Sam says, "planning a fishing trip"
Terminator flips head over heels.
Terminator says, "can i come?"
Sam smiles happily.
Sam says, "sure"
say where are you in real-life terminator?
You say, "where are you in real-life terminator?"
Terminator says, "Gold Coast"
Sam tells you, "where is that bob?"
tell sam i believe in Australia or Africa
You tell Sam, "i believe in Australia or Africa"
say where is that terminator?
You say, "where is that terminator?"
Terminator says, "Australia"
hug terminator
You hug Terminator.
say i'm afraid you can't come... we're in Michigan, USA
You say, "i'm afraid you can't come... we're in Michigan, USA"
Terminator sighs loudly.
```

Transcript 2.5: The MP transcript reads like a page off a novel (Bob's view)

[35] MP (`flysex.berkeley.edu` port `2999`) has since been renamed *ZeeMUD* and now runs at `mud.zeemud.org` port `4000`.

MP projects a user into a virtual body (or *character*) and places it (along with those of others) on a virtual world. A character is more than a 'puppet'. A user becomes embodied in his or her character. The user becomes the character.

The `look` command describes what Bob's character 'sees'. It shows that 'bob' is in a place called *Penny Lane*[36]—re-examine Transcript 2.5. The `look` command also shows that 'terminator' and 'sam' are present in the room.[37]

The `say` command allows 'bob' to speak to everyone in Penny Lane. Says will not be heard by people in other rooms. The `tell <character>` command is used to speak to a specific user.

Commands such as `smile` and `sigh` allow users to create non-verbal cues. The user 'terminator' did not have to construct a sentence to explain how 'he' felt—the `sigh` command was all that was required. Certain 'social' commands can be (or need to be) applied to another user. The `hug` command for example, requires someone to 'hug'. MP brings a hug 'alive' by describing the action in a way appropriate to each party:

When 'bob' typed the command	**`hug terminator`**
'bob' would read	`You hug him.`
'terminator' would read	`Bob hugs you.`
'sam' would read	`Bob hugs Terminator.`

The use of different pronouns heightens the sense of interaction and creates the illusion of 'physical' contact. The fact that people are actually typing in front of their computers miles from one another can become less apparent. In the author's view, MP is more than a piece of software to support online conversations—it is as Bruckman (1992) and Reid (1993) have described, a 'text-based virtual reality.'

The pretend and alias techniques can easily be employed on MP because IP addresses are protected. One simply has to choose an anonymous name for one's character to be anonymous. The administrators of MP did not require users to provide any real-life identification.

Authorship anonymity is technically possible even though the nameless technique is not supported in MP. Anonymous remarks can be created by first making one's character 'invisible'. MP (and virtually every MUD that supports invisibility) replaces one's username with the pronoun *someone* when a user cannot detect invisibility—see Transcript 2.6 on the next page.

[36] Penny Lane was one of thousands of interconnected rooms that formed the MP virtual world.

[37] The `who` command lists all the users online (ie users from every room).

```
Someone has arrived.
Someone says, "hey bob!"
look
Penny Lane
You are on Penny Lane.  Emerald Avenue is to the west and Penny Lane continues in
eastward direction.
Sam the Salmon King is standing here.
Terminator the good guy is standing here.
Obvious exits:
East  - Penny Lane
West  - Emerald Avenue
say who's there?
You say, "who's there?"
Someone falls down laughing.
```

Transcript 2.6: The username of an invisible character is replaced by the pronoun *someone* (Bob's view)

A remark tagged by *someone* has a similar effect as a nameless remark. Who is 'someone'? Which user said 'hey bob!'? If 'bob' does not know, the remark 'hey bob!' is anonymous.

The problem with invisibility is that it does not always work. Some users can detect invisibility and they can expose who is invisible—see Transcript 2.7.[38]

```
say sam who else is here?
You say, "sam who else is here?"
Sam says, "i see luke"
Someone pokes Sam in the ribs.
Someone says, "shhhhhh"
```

Transcript 2.7: 'Sam' exposing who is invisible (Bob's view)

The other concern about invisibility is that it can be used for eavesdropping. Invisibility and *sneaking* (ie the ability to enter a room without generating the usual '...has arrived' notice) are features accessible to every user. These provisions provide a way for someone to sneak into a meeting, eavesdrop on the conversation, and sneak out. These provisions give everyone a potential to jeopardise anonymity.

The author had come to learn that *wizards* (ie superusers)[39] could also become invisible. Alarmingly, a non-superuser has no way of detecting an invisible superuser!

[38] The ability to see invisible characters can be purchased using the virtual coins one gathers through the course of the game. MP is a text-based conversation system. However, it is also a multi-user computer game where users are able to pit their characters against computer-controlled characters (called *monsters*). Coins are collected from 'dead' monsters.

[39] Superusers are users that have access to certain system functions (not available by a normal user). In a way, superusers are administrators that manage a system from within the system.

To learn what 'powers' the MP superuser might have, the author examined the *MERC 2.2* source code.[40] Both MERC and MP were based on the DIKUMUD[41] source code. Two additional concerns emerged after examining the MERC code:

1 Although one's IP address was concealed from all users, it was not concealed from every superusers.

2 Although the private channel (ie tells) could not be eavesdropped by any user, it could be eavesdropped by some superusers.

In short, a MERC MUD does not adequately protect one's privacy and anonymity from superusers. The author believed the same was true about MP.

2.2.6 Foothills

Foothills (FH)[42] is another MUD. It is not a game like MP, however. Rather, its purpose is to support online conversations (just like the Internet Relay Chat and Town Meeting). Concerns about eavesdropping and invisible intruders so profound on MP did not seem to be an issue on FH.[43] FH supports lockable private rooms. Once a user locks his or her room, uninvited users cannot enter. The FH administration also gives written assurances on privacy: '...we guarantee that anything you do in private cannot be snooped or intercepted by anybody but the people you intended to talk to.'

FH allows people to choose their usernames. However, before a username can be reserved (ie protected by a password), a person will have to disclose his or her e-mail address. E-mail addresses are probably collected to prevent people from reserving multiple usernames. Whatever the reason, one will have essentially revealed one's real-life identity to the FH administration by disclosing one's e-mail address (unless one has supplied an anonymous e-mail

---

[40] The Merc 2.2 system is the work of *Michael Chastain*, *Michael Quan*, and *Mitchell Tse*. The Merc 2.2 source code was obtained via anonymous ftp from `ftp.math.okstate.edu`.

[41] The *DIKUMUD* system is the work of *Sebastian Hammer*, *Michael Seifert*, *Hans Henrik Staerfeldt*, *Tom Madsen*, and *Katja Nyboe*.

[42] `toybox.infomagic.com` port `2010`

[43] FH is based on a system called *Elsewhere II* (EW2) by *Simon Marsh*. An inspection of the EW2 source code (`ftp://toybox.infomagic.com/pub/foothills/src/EW-too.tar.Z`) revealed that EW2 does not support character invisibility (or sneaking). Of course, there is no way of confirming what FH superusers can or cannot do because the FH source code itself is not available for study.

account).[44] Fortunately, FH does not publicise one's e-mail address. The administrators will know one's e-mail address but other users will not (in theory).

Even after disclosing one's e-mail address, there is no reason why one cannot use an unregistered username when one requires a greater degree of identity anonymity. The administration should not know who is behind an unregistered username. FH also facilitates identity anonymity by concealing every user's IP address.

Authorship anonymity has also been deliberately supported by FH. The nameless technique is implemented as the `echo` command—see Transcript 2.8. This implementation of the nameless technique is also flawed because users are not prevented from making echoes when there are insufficient participants to result in authorship anonymity. Furthermore, echoes can only be made by a registered user.

```
Terminator says 'should I buy Apple or Microsoft shares?'
+ Microsoft
echo buy Apple shares while Steve Jobs is there!
+ buy Apple shares while Steve Jobs is there!
```

Transcript 2.8: Echoes are prefixed by a 'plus'

A more serious concern is the fact that superusers are able to *deanonymise echoes* (ie view echoes with the usernames attached). The problem is even more serious as the author only knew this fact because he had examined the EW2 source code. He might not have known had he not. The author does not believe it is acceptable for the administration not to warn users about this particular capability of the superusers.

The `finger` command within FH was also viewed as a potential hindrance to anonymity. The time stamps returned could be used to foil username changeovers. If 'terminator' and 'sarah' were suspected to be the same person, the time stamp of one could be compared with the stamp of the other. 'Sarah' and 'terminator' might be the same person if they were always seconds apart—see Transcript 2.9 on the next page.

[44] An anonymous e-mail account does not expose one's real-life identity in the e-mail address itself or in the header portion of an outgoing e-mail message. Page 13 described how a UNIX e-mail/shell account could be tweaked for anonymity. Alternatively, one could always supply the FH administration with the e-mail address of someone else. In recent years however, it has become increasingly simple to create an 'anonymous' e-mail account. There are numerous services on the Internet (such as *Microsoft's Hotmail*) that provide web-based e-mail accounts to people without verifying details of their identity. By supplying false personal details, one would effectively have an 'anonymous' e-mail account.

```
finger sarah
------------------------------------------------------------------------
Sarah loves to meet people
------------------------------------------------------------------------
Sarah was last seen at 09.08:14 PM-Sat, 15 February.
Her total login time is 19 days, 3 hours, 5 minutes and 12 seconds.
------------------------------- plan -----------------------------------
Get to know you better!
------------------------------------------------------------------------
finger terminator
------------------------------------------------------------------------
Terminator the newbie, so treat me nicely.
------------------------------------------------------------------------
Terminator has been logged in for 21 minutes and 38 seconds since
   09.09:10 PM-Sat, 15 February.
His total login time is 21 minutes and 50 seconds.
------------------------------- plan -----------------------------------
I must write myself a proper plan sometime ...
------------------------------------------------------------------------
```

Transcript 2.9: The **finger** command can be used to learn more about a user

More case studies will not be necessary because they will not reveal any new provision for anonymity or new concern for a user seeking anonymity.

2.3 Updated literature search

A second survey of the literature was conducted at the end of the field study. By this time, several relevant sources had emerged. Their relevance is discussed below.

Terminology

Nunamaker et el (1991) used the term *content anonymity* to describe the difficulties in attributing a specific comment to a specific person. One could say that content anonymity is the same as the author's notion of authorship anonymity.

The term *pseudo-anonymity* was used by Detweiler (1993b) to refer to the anonymity created by using a pseudonym (ie an alias) instead of one's real name.[45] Pseudo-anonymity appears to be very similar to the author's notion of identity anonymity.

[45] The word *pseudonymity* was later found in Reid 1993, Fromkin 1995, and Rigby 1995. Much to the author's surprise, *pseudonymity* was also found in various dictionaries. The author was not aware that *pseudonymity* and *pseudonymous* (the adjective form of *pseudonymity*) were two established English words.

Misuse of Anonymity

Nunamaker et el (1991), Reid (1991, 1993), Detweiler (1993a, 1993b), and May (1994)[46] have in different degrees blamed anonymity for various condemnable acts (ranging from insults to libels to extortion). While the author concedes that anonymity can be used for condemnable purposes, he does not believe that anonymity should be blamed. The author believes that anonymity is not the root of the problem. Would insults stop if people were not anonymous? In the author's opinion, 'No.' The blame should rest entirely on the people behind each act. Those that use (or rather, abuse) anonymity for condemnable purposes should be blamed.

According to Detweiler (1993a), the proper way to use anonymity is:

- not to use it frivolously,
- not to use it to provoke, harass, or threaten others,
- not to use it to evade conventions or rules, and
- not to use it where it is not welcomed by other users.

Why do people abuse anonymity? Why does Wallace (1999) write that 'the Internet has features that might unleash certain forms of aggressive behaviour in just about anyone'? The author agrees with Reid (1993) that anonymous users cannot be penalised easily or more precisely, cannot be penalised in real-life easily. People on the Internet (or rather, people conversing online) are either anonymous or geographically separated, or both. Perhaps people have been abusing anonymity simply because they could and because there were no serious repercussions.

Guidelines for supporting Anonymity

No guidelines or standards have been expressly developed to help system designers and administrators provide support for conversational anonymity. The closest to a set of guidelines (which is a set of recommendations for supporting anonymity in electronic mail and newsgroups) was also found in Detweiler 1993a.

Detweiler[47] recommended that administrators declare in writing the unacceptable uses of anonymity along with the consequences. If anonymity were abused, users should be warned. If warnings were ineffective, Detweiler suggested:

- limiting the offender's use of anonymity,
- revoking the offender's account,

---

[46] Sources that are more recent include Flinn and Maurer 1995, Fromkin 1995, Rigby 1995, Suler 1997a, and Wallace 1999.

[47] All that the author knows about Detweiler is the name *L Detweiler*. The author is unable to ascertain if *Detweiler* is in fact a real name.

- preventing access to the offender (ie forcing the offender to another service), and
- contacting the offender's local administrator or network access provider.

In Detweiler's view, the ultimate penalty for abusing anonymity was to *forfeit the offender's anonymity* (ie expose the offender's identity and actions). Obviously, this option would only be available if the offender's real identity were known. What if a user had refused to identify himself or herself (ie had adopted the alias technique)? Should such users be prevented from using a service? What if the user had provided false information (ie had adopted the pretend technique)? More seriously, what if the offender had no qualms losing anonymity?

It seemed to the author that responses other than a stern warning could only be made with the assistance of the underlying conversation system. To know which user to penalise for making a particular anonymous remark, every nameless remark might need to be logged by the system in a deanonymised form (ie with the username tags intact). To contact an offender's service provider, the system may need to keep a record of each user's IP address (or addresses). Such measures obviously compromise a user's anonymity and privacy. The author believed that Detweiler was aware of this and hence 'his' recommendation that *users be warned of any logging and monitoring activity.* Presumably, the warnings would allow users that require a high degree of anonymity to seek other services.

Another interesting demand by Detweiler was that *all programming bugs be candidly revealed*. Presumably, such knowledge would enable a user to avoid certain functions until they were fixed.

Finally, Detweiler wanted administrators to take precautions to ensure the security of the system from physical and network-based attacks and infiltration. The author agrees but adds that the administration should also *protect all user-related data from as many members of the administration as possible*. A superuser that does not need to know the IP addresses of the users should not be privy to such information.

2.4 Preliminary conclusions

How should conversational anonymity be supported? While the details are not yet clear, five ends (ie requirements) are obvious:

1 *Anonymity should be acknowledged.*
This is not simply a demand that anonymity be allowed to exist. Neither is it simply a demand that the word *anonymity* be present in the blurbs and documentation. It demands treating anonymity as a feature of the service and not simply an 'accidental' by-product of non-face-to-face electronic communication. *To acknowledge anonymity is to make deliberate changes to the system functions, standard procedures, and administrative practices for the benefit of anonymity (and the users needing anonymity).*

2 *Anonymity should not be impeded but should be simple to attain.*
This demands removing every possible impediment to anonymity.

3 *Non-anonymity (or identification) should be supported.*
This demands supporting anonymous and non-anonymous conversations. It demands making provisions to allow users to identify themselves and make identified remarks.

4 *Anonymity should be protected.*
This demands that preventable loss of anonymity does not occur. It demands protecting a user's anonymity from other users and from members of the administration.

5 *Use of anonymity should be controlled.*
This demands regulating the use of anonymity and penalising those that violate the regulations (ie misuse anonymity).

At the start of the field study, the author was concerned that he would discover a system that was perfect for conversational anonymity. By the end of the study, the author found no system that he could consider flawless, complete, or exceptional (not even the author's own Virtual-Eye system).

Town Meeting (TM) was perhaps the most ideal conversation system for anonymity. It contains more provisions for anonymity than other systems. TM acknowledges anonymity in its documentation and user-interface. Identity anonymity is easy to attain because every user's network address is protected. Since users cannot eavesdrop on one another, anonymity is further protected. In addition, administrators will not be able to add hidden functionality to TM because it is distributed as an executable binary. TM also supports the nameless technique (and hence, authorship anonymity). Going to and from authorship anonymity is a simple matter of selecting and deselecting a menu option.

Despite these provisions, the author does not regard TM as the perfect system for conversational anonymity. There are at least four problems (ie areas for improvement) in TM. First, going to and from identity anonymity is troublesome. A user's username cannot be swapped while the user is in a

meeting. A solution more akin to the `/nick` command would have avoided this problem. Second, TM supports the standard nameless technique, which has been found to be a hindrance to communication. Third, a few 'flaws' (see page 16) exist that together could jeopardise anonymity. Even if the flaws seemed trivial, they should not be acceptable for a system that has openly claimed to support anonymity. Finally, TM does not provide capabilities to prevent users from misusing anonymity or to penalise those that did.

Apart from TM (and the ViE system), other conversation systems appeared to have only *tolerated* anonymity. Anonymity was not 'supported'. It was simply achievable. The absence of face-to-face contact and use of typed text had allowed users to be anonymous. Without special support for anonymity, the path to anonymity is unnecessarily difficult.

The quest to determine how conversational anonymity should be supported is not yet over. Many specific problems have been identified and are in need of solutions. Answers that have been proposed also need elaboration and proving. In short, further research is needed. This work is described in the next chapter.

Chapter 3

In-Depth Research Strategy

3.1 Introduction

Three goals were defined for the work beyond the preliminary investigation:

1 *Create the ideal method of introducing authorship anonymity into online conversations.*
The nameless technique does not enable participants to respond to anonymous remarks in private. The *nondescript-avatar technique*[48] provides such support but it also forces a style of conversing that is far too complicated and confusing. The author intends to find a new and better method of supporting authorship anonymity than the nameless and nondescript-avatar techniques.

2 *Create the ideal environment for conversational anonymity.*
Town Meeting is believed to be the most ideal environment for conversational anonymity at present. It is not a perfect environment, however. The author intends to create a better environment for conversational anonymity than Town Meeting.

3 *Create a set of guidelines and standards for supporting conversational anonymity.*
The strategy is to transform the outcome of Goal 2 into a set of guidelines and standards that can be used by system designers and administrators to reproduce the 'idealism'.

This chapter describes the first steps towards the three goals. It describes the work that was done before the start of any practical work.

[48] The *nondescript-avatar technique* is the use of nondescript avatars (as experimented in the Virtual-Eye system) to create authorship anonymity.

3.2 Supporting Authorship Anonymity

The main requirement for supporting authorship anonymity must be that users be allowed to create anonymous remarks (ie remarks that cannot be traced to a user or person in the real-world). There should however, be six other requirements:

1 The 'normal' style of conversing is maintained

The author defines the 'normal' style of conversing as a cycle of reading remarks, typing (or replying) remarks, and waiting for new remarks. The 'normal' style of conversing is maintained by the nameless technique. The nondescript-avatar technique on the other hand, forces an unusual style of conversing.

2 Users are able to make non-anonymous (ie identified or semi-identified) remarks

Usernames were not supported for the nondescript-avatar technique to work. As such, non-anonymous remarks could not be made. The nameless technique does not prevent users from having usernames. A system simply omits a user's username from his or her remarks when anonymous remarks are required. When they are not, the user's remarks are tagged by the user's username—resulting in semi-identified remarks (or identified remarks if the user were using an identified username).

3 Users are able to make public and private (ie one-to-many and one-to-one) anonymous remarks

The nameless technique can easily accommodate this requirement. Some changes to the Foothills server for example, would allow the `echo` command to accept an additional argument (ie `echo <username>`) and hence, allow a private echo to be sent to a specific user.

4 Users are prevented from making 'anonymous' remarks when authorship anonymity cannot be attained

When there are only two participants in a meeting, each will be able to identify the remarks of the other. Authorship anonymity cannot be attained in such circumstances. A system should only allow anonymous remarks to be made when there are three or more participants in a meeting.

5 The source of an anonymous remark is only known to the underlying conversation system

A system must not enable anyone to deanonymise anonymous remarks.

6 Users are able to make private (anonymous or non-anonymous) responses to anonymous remarks

The six additional requirements essentially attempt to ensure that authorship anonymity does not hinder communication and is adequately protected. The challenge was to find a way to satisfy the six requirements simultaneously.

Simple changes to the standard nameless technique would see the first five requirements satisfied. The sixth requirement however, remained a puzzle.

A username is usually used to specify the recipient of a private message. Usernames are understood by users and by the conversation system. A remark tagged by a username tells a person where it came from. A remark addressed to a username tells the system where to deliver the remark. The system can deliver a remark as long as a destination is supplied. When a remark is not tagged with a username, a user may not know who should receive the reply. The problem with the standard nameless technique is not that the system cannot deliver the message. The problem is actually that a *user cannot specify the recipient*.

The username replacement method

In a way, a nondescript avatar played the role of a username tag. The nondescript avatars replaced use of usernames. Instead of addressing one's remarks to a username, one directed the remarks at an avatar. The ViE system was able to understand avatars just as a conventional system understood usernames. The server knew which avatar belonged to which user (or more precisely, which client). Since usernames are not used, a sender will not know which user has received his or her remark. All the avatars are identical. An avatar does not reveal whom it represents. Unlike a username, a nondescript avatar cannot be used to recognise a user. This is why authorship anonymity exists.

A nondescript avatar provides a way for a user to reply an anonymous remark in private. It provides a way to satisfy the sixth requirement. Are there other ways of replacing username tags?

Could TM's automatically generated aliases (ie *Anonymous*, *Anonymous 2, Anonymous 3*...) be used to replace usernames? Examine the following possibility:

```
say does anyone know whether i should buy Apple or Microsoft shares?
Anonymous 1 says 'does anyone know whether i should buy Apple or Microsoft
shares?'
Anonymous 2 says 'Microsoft'
Anonymous 3 says 'Microsoft of course!'
Anonymous 4 says 'buy Apple shares while Steve Jobs is there'
say who is Steve Jobs?
Anonymous 1 says 'who is Steve Jobs?'
Anonymous 4 says 'Jobs is one of the original founders of Apple computers'
Anonymous 4 says 'he left Apple a while back but he has returned to be the acting
CEO'
Anonymous 2 says 'why did he leave if Apple is that great?'
Anonymous 4 says 'dunno'
```

'Anonymous 1', 'Anonymous 2', 'Anonymous 3', and 'Anonymous 4' are 'usernames' generated by the system. They conceal the actual usernames of the four users—much like *someone* is used to conceal the usernames of invisible users in *The Mudde Pathetique*.

The idea is to use an automatically generated 'username' to specify the destination for a private message:

```
tell "anonymous 4" why are Apple shares a good investment?
You tell Anonymous 4 'why are Apple shares a good investment?'
Anonymous 4 tells you 'the share prices are a bargain'
```

Has requirement 6 been satisfied? 'Yes.' However, the use of automatically generated aliases does not create anonymous remarks. Although *Anonymous 4* is not the recipient's username, all anonymous remarks made by the recipient would be tagged by *Anonymous 4*. In essence, this method is not very different to the alias technique. An automatically generated alias is not very different from an anonymous username. Every remark tagged by a particular number will belong to a particular user.[49] Remarks tagged by an automatically generated alias should be seen as semi-anonymous rather than anonymous.

To make the remarks anonymous, a new alias (or rather, number) would need to be allocated to each remark:

```
say does anyone know whether i should buy Apple or Microsoft shares?
Anonymous 1 says 'does anyone know whether i should buy Apple or Microsoft
shares?'
Anonymous 2 says 'Microsoft'
Anonymous 3 says 'Microsoft of course!'
Anonymous 4 says 'buy Apple shares while Steve Jobs is there'
say who is Steve Jobs?
Anonymous 5 says 'who is Steve Jobs?'
Anonymous 6 says 'Jobs is one of the original founders of Apple computers'
Anonymous 7 says 'he left Apple a while back but he has returned to be the acting
CEO'
Anonymous 8 says 'why did he leave if Apple is that great?'
Anonymous 9 says 'dunno'
tell "anonymous 4" why are Apple shares a good investment?
You tell Anonymous 4 'why are Apple shares a good investment?'
Anonymous 11 tells you 'the share prices are a bargain'
```

By changing every user's 'username' continually, the remarks made each user become more difficult to identify. They become as difficult to identify as remarks that are not tagged by any username. Are 'Anonymous 4' and 'Anonymous 6' the same person? It is very likely (by looking at the content of the conversation) but there is no real way of knowing. Are 'Anonymous 3' and 'Anonymous 8' the same person? No one can really be sure (unless there were only two people in the meeting).

Authorship anonymity has been achieved because a number does not reveal anything about the user it represents. Furthermore, a number does not reveal which remarks belong to a particular user. Even if everyone knew that the

[49] Although every remark associated to a particular nondescript avatar belongs to one person, there is no way of knowing which remarks are associated to a particular avatar once the dialogue balloon moves away. Unless one has perfect memory, authorship anonymity will exist.

remark tagged by *Anonymous 4* belonged to 'Sarah Parker', Sarah's other remarks could still be anonymous because they will not be tagged by *Anonymous 4*.

The author calls this approach the *tag technique*. In theory, the tag technique has fulfilled the six requirements. It remains to be seen whether the tag technique can be implemented and whether it hides any usability problems (as the nondescript-avatar technique did).

3.3 The theoretically 'ideal' environment for Anonymity

What is an ideal environment for conversational anonymity? Is the answer in pondering what anonymous users need from a conversation system and the system's administrators? Is the answer simply a 'reverse' of the imperfections that were identified in the preliminary investigation? Is the answer the five demands described at the end of Chapter 2? The author believes that all these help to define the word *ideal*.

If the author required anonymity, he would want to be able to identify those services that deliberately support (or acknowledge) anonymity from those that simply tolerate anonymity. Anonymity is deliberately supported when a system's functions (what a system does and what it allows its users to do), standard procedures (ie what users and administrators are required to do), and administrative practices (ie what administrators are allowed to do) have been designed to suit anonymity.

The author would want to know what provisions have been made for anonymity, how they work, how they should be used, and what risks are associated with their use. The author would want the administrators to guarantee that his anonymity and privacy would be protected. He would want to know the 'powers' of the superusers.

Nothing should be allowed to jeopardise the author's anonymity. In fact, the author should be the only person capable of exposing his real-life identity.[50] If the author were to reveal his identity to the administration, the author would want the knowledge to remain confidential (ie protected). The author does not want the administration to expose his identity to any party (not even to a government agency) without his explicit permission.

[50] To *reveal or expose* one's identity is not to *jeopardise* one's anonymity. To jeopardise one's anonymity is to *reveal one's identity by accident*. A user should be allowed to reveal but not jeopardise his or her anonymity.

The author should be able to attain every 'shape and size' of anonymity using the means provided by the service. The author should not have to resort to external means to obtain the anonymity he needs.[51]

It should also be possible for the author to be identified. It should be possible for the author to embrace and discard anonymity at any time. He should also be able to switch between an anonymous and identified username without being noticed. In fact, the author should be able to make anonymous and non-anonymous remarks without any difficulty.

Finally, the author would want the administrators to provide an environment where there is law and order. When anonymity is properly controlled, it (ie anonymity) cannot be used for certain purposes. The provisions made for anonymity should not allow a user to do anything that is prohibited. The administrators need to be able to penalise those that abuse anonymity without compromising the anonymity of the 'innocent'.

3.3.1 Supporting every 'shape and size' of Anonymity

Having created and analysed who-done-it scenarios (such as the one below), the author realised that many 'forms' of anonymity exist. The notion that anonymity exists as identity anonymity or authorship anonymity has become too elementary.

Sherlock Holmes and *Dr Watson* walked towards Mrs White. 'Do you know who could have done this to Harold?' Holmes asked. 'No, I don't. Who would want to hurt my Harold? You knew him Holmes. He was a kind and gentle person.' she replied in tears. 'Well, it must be someone on board this ship,' said Watson, 'a man strong enough to overpower young Mr White.'

Holmes and Watson began to question the passengers in the adjacent cabins. They were now at Mr Lloyd's cabin. 'I saw a man leaving the room about midnight,' said Mr Lloyd. 'Do you know who he is?' Holmes asked. 'No, I couldn't see his face,' Lloyd replied.

No one else knew anything and so the housekeeping staff was summoned. They were individually questioned. Mrs Brown sat quietly—looking quite unaware of the crime that had occurred. 'What can you tell me about Mr and Mrs White, the occupants of this cabin?' Holmes asked. 'I unlocked the door for Mrs White earlier. She seemed to have locked herself out,' Mrs Brown explained. 'That's not true! She's lying!' Mrs White interrupted. 'Who is she?' asked Mrs Brown and looking very puzzled.

'We have an impostor!' exclaimed Watson. Holmes nodded in agreement. 'We have two mysterious murderers,' Holmes added, 'Mr Lloyd and Mrs Brown must have seen one each.' Mrs Brown suddenly turned pale. 'Murderers?' she asked. Holmes and Watson both gave a nod. 'Mrs Brown, come with me during

[51] As an example, the author should not have to tweak his UNIX shell account to obtain identity anonymity.

meal time,' Holmes requested, 'We'll find them.' 'I'm afraid,' was Mrs Brown's response, 'I'm sorry but I will do no such thing Sir.'

After analysing such who-done-it scenarios, the author arrived at three 'conclusions':

1 Anonymity is relative
   A person can be anonymous to one person but not to another. Mrs Brown could not recognise Mrs White—Holmes and Watson however, could. *The anonymity of one person must be spoken in relation to certain people*.

2 Anonymity is dynamic
   The more clues Holmes gathered, the less anonymous the suspects became. However, if the wrong clues were gathered, the suspects would become more anonymous.

3 Anonymity exists because of five main reasons:
   i unknown facts: where there is no information available
   ii insufficient facts: where there is some information available, but incomplete (eg missing last name)
   iii non-unique facts: where the information available (although complete) cannot be traced to a single person
   iv wrong 'facts': where false information has been accepted as the truth
   v inaccessible facts: where pertinent information (such as a person's identity) has been purposefully withheld by a third-party

Forms of Anonymity

Based loosely on the five causes[52] of anonymity, the author proposed five forms of anonymity (ie the *forms model*):[53]

1 *Absolute-anonymity*
   Absolute-anonymity is present when there is *no assessable or proven fact* about the person in question. If the murder took place in at a hotel, Holmes and Watson might only be able to guess that the murderer is someone strong. Absolute-anonymity would exist because the attribute *strong* is not an (accurately) assessable fact.

---

[52] ie unknown facts, insufficient facts, non-unique facts, wrong facts, and inaccessible facts.

[53] The order of this list does not necessarily suggest a decreasing 'degree' or 'level' of anonymity—hence the word *forms*. It is certain that there can be greater and lesser degrees of anonymity. Common sense suggests that a person hidden among ninety-nine other suspects is more anonymous than someone among two is. The number of suspects seems to be a reasonable way of measuring anonymity. Another possibility is the amount of known facts about the person. In any case, the forms model does measure or describe the depth of anonymity. It does however, provide a *consistent and concise method of describing why anonymity exists.*

2 *Profiled-anonymity*
Profiled-anonymity is present when the *facts do not lead to any suspect* (or when the *facts lead to indefinite suspects*). If Holmes and Watson only knew that the murderer is a female, profiled-anonymity would exist.

3 *Confined-anonymity*
Confined-anonymity exists when *all possible suspects are known but the person responsible cannot be determined.* Since there murderer is someone on board a ship and everyone on board the ship is known, confined-anonymity would exist.

4 *Hidden-anonymity*
Hidden-anonymity exists when *wrong facts have been accepted.* The pretend technique creates hidden-anonymity. The impostor achieved hidden-anonymity from Mrs Brown after successfully impersonating Mrs White.

5 *Protected-anonymity*
Protected-anonymity exists when a *person's identity has been intentionally withheld*. One could say that protected-anonymity exists because Mrs Brown had refused to identify the impostor.

Table 3.1 summarises the differences between the five forms of anonymity.

Table 3.1: A simple comparison of the different forms of anonymity

	Absolute-Anonymity	*Profiled-Anonymity*	*Confined-Anonymity*	*Hidden-Anonymity*	*Protected-Anonymity*
Can the (anonymous) person be recognised?	No	Yes	Yes	Yes	Yes
Number of suspects	0	0 or more	2 or more	1	1
Is the real-life identity of each suspect known?	No	No	Yes	'Yes'[54]	No[55]

Absolute-Anonymity

Absolute-anonymity is the pinnacle of anonymity. An *absolutely anonymous* person is someone whose identity (ie real name) is not known. He or she is

[54] Since the suspect has secretly assumed a false identity, no one would realise that the real-life identity of the suspect is not known.

[55] However, the real-life identity of the suspect is known to a third-party.

also someone whose *presence cannot be recognised instantly*. Table 3.2 describes how absolute-anonymity may be supported.

Table 3.2: Attaining/Supporting Absolute-Anonymity

User	*System/Service*
1 Keep one's real-life identity concealed.	1 Function without asking users to provide any real-life facts.
2 Change one's username as often as possible.	2 Allow users to choose usernames.
	3 Allow users to change usernames in secret.
3 Avoid systems (or services) that do not support anonymous remarks. Rely on anonymous remarks.	4 Support anonymous remarks.
4 Recognise one's idiosyncrasies and keep them concealed.	
5 Avoid systems (or services) that do not protect one's network address.	5 Conceal the users' network address.[56]
6 Use a method of connecting to the network that does not ask for one's real-life identity and is not exclusive to one's self or a fixed[57] group of people—eg an Internet cafe.	6 Open the system to the Internet (ie public).

Once a mysterious person can be profiled (ie recognised), absolute-anonymity no longer exists. Anything about a person that can be repeatedly observed can be used to form a profile of the person. A person's username, alias, IP address, e-mail address, and idiosyncrasies are things that allow the person to be recognised.

Profiled-Anonymity

The advantage of absolute-anonymity is greater protection. However, the advantage of profiled-anonymity is the ability to develop continuity, merit, reputation, and relationships. Table 3.3 (on the next page) describes the path to profiled-anonymity.

[56] If the IP addresses were not protected, it might be possible for one user to *finger* another user's host computer and obtain a list of online (or non-idle) users. Anonymity would now be confined to this list of people. If their real names could be determined, confined-anonymity would exist (instead of absolute-anonymity).

[57] Or confined-anonymity will result instead.

Table 3.3: Attaining/Supporting Profiled-Anonymity

User	*System/Service*
1 Anything that leads to one's real name has to be concealed.	1 Function without asking users to provide any real-life facts.
2 Use an anonymous alias as one's name and username.	2 Allow users to choose and password-protect usernames.
	3 Allow users to change usernames.
3 Conceal one's idiosyncrasies when conversing with real-life acquaintances.[58]	
4 Use a method of connecting to the network that does not ask for one's real-life identity and is not exclusive to one's self or a fixed[59] group of people.	4 Conceal the users' network address.
	5 Open the system to the Internet.

Confined-Anonymity

Confined-anonymity is created when a meeting is confined to a group of users or people. Table 3.4 describes ways to support confined-anonymity.

Table 3.4: Attaining/Supporting Confined-Anonymity

User	*System/Service*
	1 Allow users to choose and password-protect usernames.[60]
1 Avoid systems (or services) that do not support anonymous remarks. Rely on anonymous remarks.	2 Support anonymous remarks.
2 Conceal one's idiosyncrasies.	
3 Remove uninvited or unknown parties from meeting.	3 Support access restriction.[61]

[58] Idiosyncrasies would not have to be concealed among strangers.

[59] Or confined-anonymity will result instead.

[60] This allows users to identify themselves.

[61] ie support meetings lockable private rooms (such as Foothills).

Hidden-Anonymity

Hidden-anonymity is created when wrong or false pieces of information (eg a false name) have been accepted as facts. Table 3.5 (on the next page) describes how hidden-anonymity can be achieved and supported.

Table 3.5: Attaining/Supporting Hidden-Anonymity

User	*System/Service*
1 Keep one's real-life identity concealed.	1 Do not ask users to prove their identity.
2 Use a false name. Pretend to be someone else.[62]	2 Allow users to choose and password-protect usernames.
3 Fabricate idiosyncrasies.	
4 Use a method of connecting to the network that does not verify one's identity and is not exclusive to one's self or a fixed group of people.	

Protected-Anonymity

Protected-anonymity exists because one person has decided or agreed not to expose one's real-life identity to another person. Table 3.6 describes what might be needed to support protected-anonymity.

Table 3.6: Attaining/Supporting Protected-Anonymity

User	*System/Service*
1 Avoid systems (or services) that do not guarantee confidentiality of identity.	1 Conceal the users' real-life identity (from other users).
2 Conceal one's real-life identity from certain parties.	
3 Use an anonymous alias as one's name and username.	2 Allow users to choose and password-protect usernames.
	3 Allow users to change usernames.
	4 Address users by their username.
	5 Conceal every user's network address.

[62] ie the pretend technique.

Supporting the five forms of Anonymity

By combining the requirements from all the five forms of anonymity, the author arrived at a mini-framework for system designers and administrators:

1 Do not ask a user to reveal or prove his or her real-life identity.
2 Do not reveal a user's real-life identity to other users (ie do not deanonymise usernames). Users should always be address by their usernames.
3 Allow a user to choose his or her username.
4 Allow a user to decide whether to password-protect his or her username.
5 Allow a user to change his or her username.
6 Hide a user's network address.
7 Support anonymous remarks.
8 Allow users to keep unauthorised participants out of a meeting.
9 Open the system (or service) to the Internet.

If the nine directives were supported, a user should be able to choose which of the five forms of anonymity to embrace.

Levels of Anonymity

It was after the author had proposed the forms model that he came across a paper entitled *Levels of Anonymity* (Flinn & Maurer 1995). What is a 'level' of anonymity? Flinn and Maurer described how different user identification techniques create different 'levels' of anonymity—see Table 3.7.

Table 3.7: Flinn and Maurer's *Levels of Anonymity* (continued next page)

Level 5 identification (Super-identification)	▪ The identity of each user is known (to the system and administrators). ▪ No user is able to impersonate another. ▪ The system is aware of a user's activities
Level 4 (Usual) identification	▪ The identity of a user is known. ▪ Each user is assigned a single username that is protected by a password. Entry is only permitted with the right password.
Level 3 (Latent or Potential) identification	▪ The identity of every user is known. ▪ Each user is assigned a master username and password. ▪ Initial entry to the system requires the master username and password. Upon entry, a user can create and use other usernames. The user will assign a password to his or her alternate username. The alternate username and password can then be used to enter the system in the future. ▪ The system knows which usernames belong to the same person but the knowledge is hidden from other users.

Level 2 (Pen-name) identification	▪ User can have multiple self-chosen usernames—each protected by a password. ▪ The system does not know which usernames belong to the same person.
Level 1 (Anonymous) identification	▪ Entry into the system is controlled using a common access password. ▪ Users do not have usernames. Users cannot be addressed directly by the system or other users.
Level 0 identification	▪ No identification of user. 'This basically corresponds to turning on a PC that is not password protected.' ▪ The system does not keep any records of a user.

Could Flinn and Maurer's *Levels of Anonymity* model (Levels model) be used to describe a user's degree of anonymity? Is each 'level' of identification only a name or is it literally a measure of anonymity? Is a user with a lower identification level 'more' anonymous?

What constitutes greater anonymity? Common sense suggests that fewer facts are known about a more anonymous user. In other words, a non-registered user Foothills user should be more anonymous than a registered user (since a registered user's e-mail address would be known). For the Levels model to agree with common sense, a non-registered user would need to have a lower level of identification (or higher level of anonymity). According to Table 3.7 (above), a registered Foothills user would have a Level 3 identification. A non-registered user would have a Level 2 identification. In this instance, the Levels model appears to agree with common sense.

Further investigations revealed that it is not always possible to match a user or system to a particular 'level' of identification. A system such as Town Meeting (TM) does not support protected usernames but does provide the ability to restrict access to a meeting via a common access password. A TM user cannot be classified under Level 2 identification since usernames cannot be reserved (ie since password-protected usernames are not supported). Level 1 identification will not be correct even though a common access password is supported because every user has a username. Perhaps more levels are needed in the Levels model.

The Levels model also fails to consider the anonymity among users. If user A were to know the identity of user B but not user C, C would obviously be 'more' anonymous than B. B would be 'less' anonymous because the identity of B would be known to A and could be jeopardised by A (by accident or otherwise). To gauge a user's level or degree of anonymity more justly, two perspectives must be considered:

1 how much the system and administrators know about a user, and
2 how much other users know about a user.

The Levels model only considers the first point.

Measuring the potential for loss of Anonymity

The author believed that a 'better' way to gauge a user's depth of anonymity was to measure the *anti-anonymity factors* (ie factors that could jeopardise the user's anonymity). He believed that a user exposed to fewer 'risks' (ie fewer anti-anonymity factors) would be someone more anonymous.

A user's anonymity is 'lessen' (or at risk) when:

1. the user has disclosed *true* personal details, either real-life (eg one's real name or address of residence) or virtual (eg one's e-mail address) to the administration.
2. the information submitted by the user is *verified* in some way—eg a user is asked to send an e-mail from the e-mail address submitted.
3. the user makes *long-term* use of a username—enabling others to develop a profile of the user.
4. the user's identity is known to another user—'doubling' the danger of slip-ups.[63]
5. the user's idiosyncrasies are unique and obvious.
6. the system is only open to a specific group of people—ie the participants are known.
7. the user had 'logged on' directly from his or her private computer at home instead of a multi-user host or a public computer at an Internet cafe.

Circumstances that can 'enhance' or increase a user's anonymity (ie pro-anonymity[64] factors) are:

1. when the system keeps a user's personal details (eg IP addresses) and activities confidential from (or inaccessible to) other users.
2. when the system keeps a user's personal details and activities confidential from (or inaccessible to) administrators.
3. when there are many participants and a high participation rate.[65]
4. when private communication is not supported. Limiting participants to public comments helps to prevent participants from *corroborating* (ie exposing their identity to one another) in secret. If one knew three out of four participants in a meeting, one might be able to work out who was using the fourth username.
5. when authorship anonymity and identity anonymity are supported (or possible).

---

[63] Instead of one, there are now two people that could make the slip-up.

[64] The absence of a pro-anonymity factor could be seen as an anti-anonymity factor.

[65] Thereby, increasing the number of decoys and the chances of misprofiles.

The Anti-Anonymity Checklist 1

Most of these factors have been put into what the author calls the *Anti-Anonymity Checklist 1* (or AAC1). The AAC1 is a list designed to gauge the amount of anti-anonymity factors associated to a specific user in a particular situation—see Table 3.8.

Table 3.8: The Anti-Anonymity Checklist 1

	Factors	*Points*
1	A username can be protected by a password: 1 point. A user uses a password-protected username: 2 points.	1–2
2	A user has provided (ie submitted) true real-life information to the system or an administrator.	2
3	Administrators have access to the user's real-life details.	2
4	User's privacy can be compromised.	2
5	User's personal details are made public.	2
6	Remarks are tagged by user's username.	1
7	Anonymous remarks can be deanonymised.	1
8	User idiosyncrasies are recognised.	2
9	Private communication is supported.	1
10	Meeting or conversation can be restricted to a specific group of people: 1 point. Real-life identity of the group is known (ie a closed meeting): 2 points.	1–2
11	Participants connected from the same locale (ie room or building).	2
12	User's real-life identity is known to one or more participants: 1 point. One or more participants have not guaranteed to keep the user's identity confidential: 2 points.	1–2

Weights were assigned to each factor—two points were assigned to a factor that could lead to loss of identity anonymity. These points could be totalled and converted to a percentile. A higher percentage should suggest that a user is less anonymous. Of course, a rating of 100% should not suggest that a user is not anonymous. It should only indicate that the user is less anonymous than one with a lower percentage. Similarly, a rating of 0% does not suggest that anonymity cannot be lost.

The AAC1 checklist should allow the author to make simple comparisons between various circumstances and systems. Consider the following example. *Sarah Parker* used the pretend technique and became 'Peter Parker'. She chose the username 'peter' to reinforce her deception. Assume that 'bill' knew 'peter'

was Sarah but had agreed to keep it a secret. What AAC1 rating did Sarah possess?

Table 3.9 shows that Sarah has an AAC1 rating of 39% on a typical Internet Relay Chat (IRC) server.

Table 3.9: AAC1 analysis of Sarah on IRC

	Factors	*Points*
1	Password-protected username is not supported.	0/2
2	Personal details submitted? No.	0/2
3	Admin knows Sarah's IP address.	1/2
4	Compromised privacy? No.	0/2
5	Lack of Confidentiality? IP addresses are exposed.	2/2
6	Remarks tagged by username? Yes.	1/1
7	Deanonymisation of anonymous remarks? Anonymous remarks not supported.	N/A
8	Known Idiosyncrasies?	N/A
9	Private communication supported? Yes.	1/1
10	Meetings can be restricted. Closed meeting? Assume no.	1/2
11	Proximity? IRC is open to the Internet.	0/2
12	Sarah is not anonymous to some participants? Yes, but 'bill' has agreed to provide confidentiality.	1/2
	Total	7/18
	Percentage	39%

On Town Meeting (TM), Sarah would have a rating of 34%—see Table 3.10 (below)

Table 3.10: AAC1 analysis of Sarah on TM (continued next page)

	Factors	*Points*
1	Password-protected username is not supported.	0/2
2	Personal details submitted? No.	0/2
3	Admin has access to personal details? No.	0/2
4	Compromised privacy? Yes, The TM server logs all non-private remarks.	1/2
5	Lack of Confidentiality? No.	0/2
6	Remarks tagged by username? No, the nameless technique is supported.	0/1

	Factors	*Points*
7	Deanonymisation of anonymous remarks? A flaw in TM may allow people to deanonymise anonymous usernames (and remarks).	0.5/1
8	Known Idiosyncrasies?	N/A
9	Private communication supported? Yes.	1/1
10	Meetings can be restricted. Closed meeting? Assume no.	1/2
11	Proximity? TM is not open to the Internet.	2/2
12	Sarah is not anonymous to some participants? Yes, but 'bill' has agreed to provide confidentiality.	1/2
	Total	6.5/19
	Percentage	34%

What do these figures suggest? Did Sarah have a 39% 'probability' of exposure on IRC (and 34% on TM)? There is no evidence to support or deny such a claim. Is Sarah more anonymous on TM than IRC? 'Yes,' according to the AAC1 ratings. It also makes sense because one's IP address (which is exposed on IRC) can be traced to one's personal computer. How significant is the 5% difference? The author does not know.

What if 'bill' did not know that Sarah was 'david'? Common sense suggests that Sarah would now be 'more' anonymous (or less at risk). The IRC ratings dropped (by 5.6%) to approximately 34% while the TM ratings dropped by (5.3%) to approximately 29%. These drops in percentages correctly reflect the new circumstance. However, how significant is the additional 0.3% drop experienced on IRC? Why would the removal of 'bill' provided more benefit on IRC?

Common sense suggests that if 'bill' had exposed the identity of 'david' on IRC, it would have been possible for people to identify Sarah's presence even if she were to use a new username (because Sarah can be recognised by her IP address). On the other hand, if 'bill' had exposed 'david' on TM, Sarah could simply used a different username to regain some of the identity anonymity lost. In other words, 'bill' was a greater threat to Sarah on IRC. Hence, the removal of 'bill' would have caused a greater drop (in risk) on IRC. Perhaps the AAC1 is more accurate than the author has expected.

The accuracy of any anti-anonymity factor checklist will depend on the factors and correct weight assignment to each factor. The more comprehensive a checklist, the more accurate the measurement should be. Is accuracy important? What is the value of knowing that 'david' might be 5% 'more' anonymous (or 5% more protected) on TM? While absolute accurate is always important in any scientific study, the author believes it is not possible with anonymity. The author's own belief that anonymity is dynamic and relative tells him that anonymity cannot be measured with absolute accuracy. By using the same checklist for analysis, relative accuracy should be possible.

The author sees the AAC1 percentages as indicators for comparison rather than probability predictions. If the AAC1 was able to show that one environment was 'safer' than another was, it has already served a useful purpose.

3.3.2 Strategies for protecting Anonymity

Realising that users can be ignorant (ie forgetful or careless), system designers and administrators should remove every possible danger to anonymity. If a particular danger cannot be removed, users should be warned about the danger *before* they are allowed to use the service. Such warning can be included in the promotional materials, sign-up instructions, or login process.

What dangers to anonymity can be removed? A system can be built to detect certain keywords. Multi-User Domain (MUD) systems will usually allow a user to specify the gender for his or her character (ie virtual body). On such systems, pronoun checks could be performed on a user's remarks. If the wrong pronoun were found, the system could alert the user and allow the user to make a correction if needed. Alternatively, the system could allow a user to specify a set of keywords to detect. Such a provision would allow simple slip-ups and idiosyncrasies[66] to be avoided.

Flinn and Maurer (1995) suggested the idea of automatically modifying a user's (or every user's) remarks to a specific style of writing. Such a provision (which Flinn and Maurer called *style scramblers*) would certainly benefit authorship anonymity. When all remarks appear to have the same style of writing, it may be impossible to determine which remarks belong to which user.

According to Flinn and Maurer, the style scrambler could also be used to create distinctly different styles of writing. This should be particularly useful to the pretend technique. If one were to pretend to be someone 'uneducated', the style scrambler could be made to introduce grammatical errors on purpose. Of course, Flinn and Maurer have yet to demonstrate that their ideas are implementable. It is certain however, that technologies to correct spelling and grammatical errors in one's remarks already exist.

Another danger that is easily removed is the 'two-person' problem. A system can prevent a user from using an authorship anonymity technique (such as the nameless or tag technique) when there are fewer than three active users in a meeting. Users that could not have possibly made a particular anonymous remark cannot be counted as an active user.

One's anonymity will also require protection from other users. If 'bob' were to know the identity of 'terminator', 'bob' could jeopardise the anonymity of

[66] If one had the habit of including a smiley :) in one's remarks, the smiley could be specified as one of the keywords to filter.

'terminator'. What could 'terminator' do if 'bob' were to expose the real name of 'terminator'? There are two obvious options:

1 'Terminator' can resort to an authorship anonymity technique (such as the tag and nameless techniques) to attain authorship anonymity (ie to make anonymous remarks). Even when people know who 'terminator' is, the remarks made by 'terminator' can remain mysterious (ie anonymous).

2 'Terminator' can assume a different username—one that is not known to 'bob'.

How did 'bob' know the identity of 'terminator'? How did 'terminator' lose identity anonymity? Could that loss have been prevented? There are at least four ways 'bob' could have known that 'terminator' was 'Sarah Parker':

1 Sarah revealed her real name to 'bob' (or someone told 'bob' that *Sarah Parker* was the real name of 'terminator').

   The author believes there is little the administration or system can do to prevent this possibility. Just as a user should not be deprived of anonymity, the user should also not be deprived from being identifying himself or herself (especially when it does not jeopardise the anonymity of another user). The administration can however, help a user who had lost (or given up) anonymity to gain it back. Sarah (ie 'terminator') should be allowed to change her username. The change should be done with minimal difficulty and repercussion. Sarah should be able to retain all her 'privileges' (or settings) she had while she was 'terminator'.

2 'Bob' eavesdropped on a conversation where 'terminator' revealed her identity to someone else.

   Avoiding this possibility simply requires the removal of all eavesdropping provisions. A user should not be able to conceal his or her 'attendance' in a conversation. A user should not be able to intercept any message not addressed to the user. The author believes that these requirements should also apply to the system's administrators.

3 'Bob' traced the username *Terminator* to an identified username (eg to 'sarah_parker').

This could have happened when Sarah was swapping usernames. Two users sharing the same network address might reveal that both are the same person. The same would happen if two users shared the same idiosyncrasies. Username changes can also be foiled if a user did not include a 'reasonable' delay.[67]

A system can help to make username changes more 'successful' by:

- concealing the network address of users,
- not arranging a participants-list in the order in which users had entered the system (or meeting),
- not 'time-stamping' a user's entry and exit, and
- not informing other users that a user has left or entered the system (or meeting).

The last suggestion seems to go against the earlier recommendation that no one be allowed to 'sneak' into a meeting. If one could sneak into a meeting, one could potentially eavesdrop on a conversation. In view of that, perhaps it should not be considered.

The problem with idiosyncrasies is more complicated. The author has repeatedly demanded that one's idiosyncrasies be recognised and hid. It may also help if *one were to fabricate a set of 'idiosyncrasies' for each of one's usernames*. By exhibiting different 'idiosyncrasies', one may be able to persuade others to believe that a different person is behind each of one's usernames.

4 'Bob' was able to use a flaw (or provision) to access personal details on 'terminator'. Alternatively, a member of the administration might have given certain (privileged) information to 'bob'.

This possibility can be avoided if real-life information is not requested or recorded. Is it possible to operate a service without knowing the users' identity? It is certain that the task of capturing data can be fully automated. A user should not have to provide information through an administrator or operator. The user should be able to enter any required data on his or her own. Verification of any data should also be automated. In fact, the operations of a service should be automated as much as possible. The rationale is that *a system can be engineered (ie forced) to maintain perfect confidentiality whereas people cannot*.

Whatever is known about a user must be kept in confidence. Nothing should be disclosed to a third party without the user's explicit consent. Administrators of every level need to respect or be taught (or *forced*) to respect privacy and anonymity.

[67] A participant that joins a conversation just after (or before) one leaves may expose the fact that the two are the same person.

3.3.3 Operating a service with Anonymous users

Knowledge of each user's identity is important for managing resources, offline communication, and accountability. How would these functions be performed if the identity of the users were not known?

Names are an important resource—perhaps the most important in a conversation service. If not managed properly, people would not be able to have the username they desire. One way of ensuring that all the 'popular' names have not been used is to limit the number of names a person can reserve (ie protect by a password). There would certainly be more unused names if each person were only allowed to reserve one username.

How would such a policy be policed? How would the administrators tell that two usernames have been reserved by one person? The author could not find a definitive answer. An alternative to preventing users from reserving multiple usernames is the *deletion of usernames that are not in regular user*. Two usernames should be freed (for others to use) if they have not been used for a period of time. It does not matter whether both usernames belonged to one or two different people. A user that does not have the time (or no longer wants) to use a service should be removed.

Disk space may be another issue for concern. The amount of data stored about each user can vary from system to system. In the case of MUDs, an exceptionally large amount of data may be stored. This is because each reserved username (ie 'character') is essentially a virtual body. Each 'body' can be equipped with various 'armour' and 'weapon'. Details of each character's equipment have to be recorded.

To discourage players from over-equipping their characters, a *'fee'* is usually charged. It is usually proportional to the size of a user's file (or rather, to the number of items collected). The more equipment a user amasses, the larger the user's file gets.[68] The larger the file, the more the user has to pay. A user's equipment will not be saved if the user does not have sufficient virtual coins.[69] A user that does not have the time to play will not have enough coins to retain his or her equipment. This simple economic burden provides a way to keep the size of files from growing unreasonably large. Such a burden can also be used to delete a user's file altogether (and free the username).

Network bandwidth is another important resource. As the number of online users increase, the size of the bandwidth available to each user shrinks. As transmission rates drop, remarks begin to take longer to appear. Connections begin to be broken because the client and server cannot verify each other's existence.

---

[68] Even though the amount of free disk space has become less of a concern (as large capacity hard drives are now becoming very affordable), it remains a fact that a larger database or file will take longer to access (ie read into memory). On a slower hard drive, a whole computer may freeze while data is read.

[69] Users earn 'coins' by taking them from defeated 'monsters'.

Bandwidth is wasted when users leave multiple characters online (ie multiplay). Even when a user's characters are not generating remarks, bandwidth is still being used because (public) remarks will need to be transmitted to the user's computer. If each user were to establish two connections to a server (ie have two online characters), the network bandwidth would be unnecessarily reduced. Again, how do the administrators know which characters or users are the same person in real-life? There is no clear solution. Again, the answer may be to *disconnect* idling users. Two connections should be severed whether both belong to one person or two different persons. The rationale is that a user that does not want to converse with others should not be on a conversation system.

When network performance drops below an acceptable level, an administrator may choose to 'lock up' a system.[70] Users that attempt username changes[71] at such a time may find themselves unable to re-enter a system. Even if a system were only closed to visitors, it would affect anonymity because one might not be able to create and use an anonymous username—see Transcript 3.1.

```
Please enter your name:zorro

        ----------=>>>>> Foothills <<<<<=====-----

      Sorry, this program is temporarily closed to new players.
      Please try again soon, or if you wish to have a character
                     registered send email:
      specifying a character name, the password you wish to use,
           and your email address to the address below.

            fha@toybox.infomagic.com

      Please mail any comments or questions to those addresses.
```

Transcript 3.1: A system may prevent one from creating an anonymous username

If a user's e-mail address is not collected, how will the administrators be able to communicate with an offline user? *'Passive' communication* may be the answer. Instead of sending a piece of information to a user, the administrators can leave a piece of information for the user to collect. A website (or bulletin board) can be established to communicate general news. An 'e-mail system' (where a user's username will suffice as the 'e-mail address') can be integrated into the conversation system. This will enable a messages to be left for a specific user.

While no personal information may be requested from users, certain users may volunteer information. Some information may also be freely available. A user's network address for instance, can be obtained from the network

[70] Of course, network performance is not the only reason why a system may be closed. The next subchapter (ie Ch 3.3.4) will explain other reasons.

[71] To switch between anonymity and non-anonymity.

protocol. How should such 'free' information be handled? The author believes that every piece of information about a user should be treated as confidential. If possible, no information should be recorded. Storage and use of personal information should be *transparent*. Anything that is recorded about a user should be revealed to the user. A user should also be told who has access to which piece of information (about the user).

3.3.4 Strategies for controlling Anonymity

How can the administrators penalise (anonymous) users that abuse anonymity? A solution may be to demand that every user identify himself or herself before being allowed to use the service. In return, the administrators would have to assure users that their identities would be concealed from one another. By knowing the real-life identity of a user, real-life actions can be taken. This possibility or 'threat' may be sufficient to discourage users from abusing anonymity.

A 'better' solution may be to grant anonymity to (or rather, protect the anonymity of) a user as long as the user abides by certain conditions. Should a user break any rule, the administrators would have the right to deny the user access to anonymity (eg prevent the user from creating anonymous remarks), or as Detweiler (1993a) suggested, forfeit the protection given to the violator (eg expose the user's IP address).

Is it possible to penalise a user without compromising or reducing his or her anonymity? Banishment (or 'denial of service') is the strategy adopted by many public conversation services. A system can be designed to reject network connections from a specific computer or site. Such bans can be imposed without exact knowledge of the violator's identity.

The author believes that an administrator should only need to supply the violator's username and specify the type of ban to impose. An administrator should not need to know the violator's network address—the system should know the network addresses of every user.

A system should be designed to allow the provision for creating anonymous remarks or new usernames to be completely disabled. Where there are no known 'suspects', these fundamental provisions for anonymity can be disabled.

Another strategy may be to equip users with the ability to *insulate* themselves from anonymous users. ElseWhere II (EW2) systems such as Foothills allow users to block private communication from specific users (or from all users). Furthermore, registered users are able to move their conversations into their private rooms. These rooms can then be locked to keep strangers out.

Another way of controlling anonymity is to *confine* the ability to create anonymous remarks to certain rooms or events (eg a brainstorming session). A system may for example, equip a chairperson (or host) with the ability to control when anonymity is permitted and when not.

The strategies, techniques, and ideas that have been proposed in this chapter represent the beginnings of a framework for supporting conversational anonymity. The task ahead is obvious—to implement the proposed provisions. That is described in the next chapter.

Chapter 4

Implementation

4.1 Introduction

Can the proposed provisions outlined in Chapter 3 be implemented? Will the provisions work together? These were the challenges for the author.

A conversation system was needed to serve as the foundation for implementing the proposed provisions. The MERC 2.2 source code was chosen not because of any single compelling reason. There several 'good' reasons, however:

1 The MERC system did not provide any special support for anonymity—enabling the author to make fair comparisons between the 'naturally occurring' form of anonymity and a deliberately supported form.

2 The MERC source code (written in C) could be compiled without any change on Ultrix,[72] which was the operating system of the host computer available for the author to conduct his research.

3 The author believed that MERC (and most combat-oriented MUDs) could offer a 'superior' kind of interaction—one that was similar to meeting face-to-face.

MERC is an adventure-oriented multi-user domain (MUD) system (like *The Mudde Pathetique*). It did not bother the author that MERC was a game. He believed the 'play' element could in fact, be used to attract people to the system.

Oz was the name given to the resulting (MERC) MUD after it had undergone a series of modifications. Oz was still an adventure MUD. However, it was now providing some support for anonymity. Everything that was needed to introduce the tag technique was implemented on Oz.

[72] Digital Electronic Corporation's version of the UNIX operating system.

4.2 Implementing the Tag technique

Oz needed to assign numbers to anonymous remarks. It needed to know which number belonged to which user. Users needed a way to specify whether remarks were to be tagged by their username or with a number.

New communication commands

A set of commands (ie the *tag commands*) was added to enable the tag technique to be used—see Table 4.1. The tag commands were designed to resemble their non-anonymous counterparts. From a user's point of view, communicating anonymously simply required the addition of the prefix *A*[73] to an existing communication command—eg `asay` instead of `say`.

Table 4.1: Commands for using the tag technique

Syntax	*Purpose*
`asay <remark>`	Send an anonymous message to everyone in the same room. A number replaces the sender's username.
`atell <character> <remark>`	Send an anonymous message to a single user. A number replaces the sender's username.
`atell <number> <remark>`	Send an anonymous message to a mysterious recipient. A number replaces the sender's username.
`ashout <remark>`	Send an anonymous message to everyone online. A number replaces the sender's username.
`tell <number> <remark>`	Send a non-anonymous message to a mysterious recipient.

When a user makes an anonymous-say (ie an asay), Oz would allocate a number to the user.[74] This number would take the place of the user's username—see Transcript 4.1 (below). A maximum of twenty tags ensured that users would only have two digits to type. Once '20' was used, '1' would be reallocated (ie reused).

```
asay Does anyone know the way to ultima?
20: Does anyone know the way to ultima?
asay doesn't anyone know?
1: doesn't anyone know?
ashout Does anyone know the way to ultima please?
You (anonymously shout): Does anyone know the way to ultima please?
3 (to you): What do you have to offer in return?
```

[73] *A* for *Anonymous*.

[74] The relationship between a tag and a username would be maintained in memory. This information would not be recorded on any file. It would not be accessible to any user or superuser.

```
atell 3 i have 100000 coins for you
You (anonymously tell 3): i have 100000 coins for you
5 (to you): OK, I'll meet you in 5 minutes at the boar inn
```

Transcript 4.1: The tag technique creates remarks that are tagged by numbers instead of usernames

When a remark is directed to a number, Oz will look up the *tag-username records* to convert the number into a username. Oz will then proceed to find an online user that matches the username. Once a match is found, the message is delivered.

What would happen if a remark were directed at a number that had not been assigned to anyone? What if a number referred to a user that had left the system?

Undeliverable Anonymous remarks

MERC (and therefore, Oz) had been designed to generate an error message when a remark cannot be delivered. Such feedback was believed to be detrimental to anonymity. It might allow a person to determine the user behind a particular number (or more precisely, a particular remark). Someone that understood the tag technique could guess that '7' was 'sue'—see Transcript 4.2.

```
ashout Does anyone know the way to ultima please?
You (anonymously shout): Does anyone know the way to ultima please?
3 (to you): what do you have to offer for that information?
atell 3 i have 100000 coins
You (anonymously tell 3): i have 100000 coins
5 (to you): ok, i'll meet you in 5 minutes at the boar inn
atell 5 who are you? how will i know you?
You (anonymously tell 5): who are you? how will i know you?
7 (to you): i'll come on as iris
atell 7 oh, ok
You (anonymously tell 7): oh, ok
Sue has left the game.
atell 7 oh yeah and I'll be ZZZZ
Sorry, that person is no longer online.
```

Transcript 4.2: A delivery error may reveal that '7' was 'sue'

The problem is compounded further by the fact that 'sue' will not know that 'she' has been 'discovered' (ie that she has lost authorship anonymity). The author believed that any loss of anonymity if avoidable should be avoided. If delivery errors were not reported, perhaps one would be left to wonder whether the recipient:

- did not want to respond,
- was still thinking about a reply, or
- was no longer online.

The author believed that leaving a user uncertain (or frustrated) was preferable to putting someone's anonymity at risk. Oz was consequently made to suppress error messages when a remark could not be delivered to a number (ie to a mysterious recipient).

The decoy checker

There were other ways of foiling the tag technique. The tag technique would not work if there were insufficient people to act as one's decoy. Decoy checks were introduced to ensure that the tag technique could not be used unless there were more than *three* participants.[75]

The checks involved more than ensuring that there were three users in a room. Transcript 4.3 shows that there are five users (including 'terminator') at the *Entrance to the Grunting Boar Inn*. Why did the author (or rather, the decoy checker) prevent 'terminator' from making an asay?

```
look
Entrance to the Grunting Boar Inn
You are standing in the entrance hall of the Grunting Boar Inn.  The hall
has been wisely decorated with simple but functional furniture.  A small
staircase leads up to the defunct reception room and the bar is to the east.
Sue the Cowgirl is here.
Ally Cat is sleeping here.
Bob is JR's brother! is sleeping here.
(Invis) Luke Skywalker is here.
asay does anyone know the way to ultima?
Sorry, can't do that right here. Too few people around to be anonymous this way.
```

Transcript 4.3: The *decoy checker* stopping 'terminator' from making an asay

The asay was not permitted because the decoy checker had only detected two valid parties in the room—ie 'sue' and 'terminator'.

The users 'ally' and 'bob' were not considered because their characters were 'asleep'. A user cannot make an asay (or any other remark) while his or her character is 'asleep'.

The user 'luke' was not counted because 'his' character was invisible. Consider this scenario. If 'sue' were not able to see invisible characters, 'she' would not know that 'luke' was present. If the decoy checker counted invisible characters, there would have been three valid parties (ie 'terminator' and 'luke' and 'sue'). Asays would have been possible. If 'luke' had made an asay, 'sue' might be led to believe it was made by 'terminator' (since 'she' would not know that 'luke' was present). The user 'terminator' would have become a 'scapegoat' instead of a decoy. If 'sue' were able to see 'luke', she would have traced the asay to two users—'terminator' and 'luke'. She would not have jumped to any

[75] A gathering of three users is the theoretical minimum—ie one speaks while the other two serve as a decoy for each other.

conclusion. The author did not want to give anonymity to one user at the expense of another. To avoid such 'injustice', invisible characters were not counted.

The decoy checker would also ignore characters that were not under any human control. Any character listed by the `look` command but not by the `who` command (ie is on the *look-list* but not the *who-list*) would be a computer-controlled character[76] or a *link-dead* character. When a user exits Oz 'gracefully', the user's character is removed from a room. However, if a user were to be unexpectedly disconnected (because of network problems), the user's character would not be immediately removed. Such a character is called link-dead.[77] A link-dead character is not under anyone's control and could not have made any remarks.

Although the decoy checker prevented 'terminator' from making an asay (in Transcript 4.3 on the previous page), it does not mean that anonymity cannot be attained. It simply meant that 'terminator' could not use the tag technique to attain anonymity. The user 'terminator' would be able to make semi-anonymous remarks if the username terminator were anonymous.

The problem created by multiplaying

The decoy checker works on the assumption that a different character is controlled by a different person. In reality, one person could be controlling two or more characters. It is possible that a meeting among four users is really a meeting between two people—one person may be controlling three characters. In such a scenario, the decoy checker would have (wrongly) concluded that there were sufficient participants.

The rules of Oz clearly stated that no one should use two characters at a time. However, there was no simple way of policing this rule. The author could not identify people that were multiplaying nor prevent multiplaying from occurring. Identical IP addresses could certainly be a sign but not necessarily proof of multiplaying. People connecting from one common UNIX host for example, would have the same IP address.

Again, this problem could be solved if Oz knew the (identity of the) person behind every character. If that were possible, Oz will know which usernames belong to the same person. Oz would be able to prevent a person from bringing a second character into the virtual world. There are several 'problems' with this solution, however. First, how will users prove who they are? Would users be willing to provide information such as their credit card

[76] These computer-controlled characters are part of the game. When 'killed', they provide users with coins, points, and equipment.

[77] Oz would eventually remove link-dead characters from the landscape. A link-dead character is not immediately removed because that would enable the character to continue 'playing' (eg 'fighting') while its owner made attempts to re-establish connection.

number or passport number? Second, would the author be able to verify any of the information provided? Since Oz was not operating as a pay service, such verification methods were not feasible.

Could e-mail addresses be used to identify a person? In theory, 'no'. Web-based e-mail services such as *Microsoft's Hotmail* (`www.hotmail.com`), *Yahoo's Yahoo! Mail* (`www.yahoo.com`), and *Netscape's Webmail* (`webmail.netscape.com`) were giving away e-mail accounts without verifying that people were who they claimed to be. In other words, a person could easily have several anonymous e-mail addresses.

How was multiplaying addressed then? The author found his 'solution' when assessing the number of characters a person could realistically control. The author recalled instances (during the field study) where he could successfully control two characters at time. In fact, the author could even engage in two different conversations simultaneously (where each character was engaging in its own conversation). Playing the role of two people was mentally stressful but not impossible. Could someone manage three identities (ie characters) simultaneously in one meeting? The author tried but could not manage three conversations in three windows. Of course, this does not mean it cannot be done. Even if it were possible, what would drive someone to attempt such feats? The author believed that raising the minimum number of participants (required by the decoy checker) from three to four was a sensible way to address the problem created by multiplaying. Although this change was not necessarily a solution, it would remove the problem of multiplaying among three users. For multiplaying to foil authorship anonymity, a person would now have to play the part of three different people at a meeting.

If users were to use the alias or pretend technique in conjunction with the tag technique, multiplaying would not be such a menacing problem. Should a user be robbed of authorship anonymity, the 'exposed' (ie deanonymised) remark would only be traced to an anonymous username or a false name (ie to a fictitious person). Should a user be robbed of authorship anonymity, he or she would still be left with identity anonymity.

The problem of multiplaying could be removed if users were to conduct *closed meetings*. There are no anonymous users in a closed meeting. Every user in a closed meeting would be a different person. Could Oz support closed meetings?

For closed meetings to be possible, a group of people need to be able to identify themselves, and keep anonymous or uninvited users out of the meeting. The first requirement was possible since users could create password-protected characters. No one could use a particular password-protected username without knowing the proper password—no one could be 'terminator' except 'Sarah Parker'. The second requirement could not be met,

however. Oz did not provide a reliable way of keeping uninvited guests (including superusers)[78] out of a room (ie meeting).

Lockable chambers

Lockable chambers were introduced to support closed meetings. Every user has access to a lockable chamber. The `chamber` command will bring a user into his or her chamber. The `chamber lock` command is used to lock and unlock one's chamber—see Transcript 4.4.

```
chamber
You see a chamber belonging to Terminator.
Isn't it time for a description?
Sue enters for a visit.
Sam enters for a visit.
Sam smiles happily.
Gates enters for a visit.
Sue says 'hello Terminator!'.
say glad all of you could come
You say 'glad all of you could come'.
chamber lock
*Click* Your chamber is locked.
```

Transcript 4.4: A chamber can be locked once the expected participants have arrived (Terminator's view)

Once locked, no one (including superusers) will be able to enter one's chamber.[79] Once locked, the gathering would become a *private meeting*. If 'terminator', 'sam', 'sue', and 'gates' knew each other's identity (ie real name), the gathering would have become a closed meeting.[80]

To prevent disruptions to a closed meeting (or any private meeting), the `earmuff` command was introduced. Anyone with earmuffs activated would not receive non-local communication such as shouts and tells—see Transcript 4.5[81] below.

```
tell sam where is Luke?
Sam doesn't want to be disturbed.
Sam says 'let's have our earmuffs on'.
```

[78] MERC (and therefore, Oz) allowed a superuser to enter any and every room in the virtual world.

[79] The 'line-tapping' (ie the `snoop <character>`) command was also removed from superusers. If it were not, the head superuser would have been able to intercept the transmissions between Oz and someone in the locked chamber (eg the owner of the chamber). The head superuser would have been able to read most of the remarks made in the locked chamber.

[80] Asays made in a private or closed meeting should create what the author calls confined-anonymity (see page 42).

[81] With earmuffs on, one would not receive ashouts and atells as well.

```
nod
You nod solemnly.
earmuffs
Earmuffs are on. You will stop hearing tells and shouts.
say where is Luke sam?
You say 'where is Luke sam?'.
```

Transcript 4.5: Earmuffs and a locked chamber essentially allow a group of people to isolate themselves completely from everyone else (Terminator's view)

4.3 The Oz experience

Oz[82] was now fully operational and ready for 'real' use:

```
                              Welcome to Oz

          Oz ...  ... . . .   Oz...  .. . .......Oz........oz......

     Oz was created from the Merc (diku mud) code by Kahn, Hatchet, Furey.
          Diku mud originally by Hans Henrik Staerfeldt, Katja Nyboe,
               Tom Madsen, Michael Seifert, and Sebastian Hammer.
                              Oz code by Drew.

    By what name do you wish to be known?
```

The initial group of users were people that the author had invited from *The Mudde Pathetique*. The author knew many of them personally—even their real-life identity. Some of these people became superusers on Oz. The 'distinguished' role of the head superuser was (obviously) filled by the author. This position gave the author the opportunity to put anonymity to use during the 'staff' meetings.

4.3.1 The first six months of operation

Superuser meetings were held at least once a month. The tag technique proved to be particularly useful in these meetings because it enabled anonymity to be introduced without the use of anonymous usernames. This was important since the author needed the superusers to use their official usernames for identification. The author did not want any non-superuser to be present in those meetings.

For most of the time, the author was not able to tell the origin of the *numbered remarks* (ie remarks anonymised by the tag technique). On

[82] 131.244.15.53 port 2000

occasions however, he was able to recognise certain idiosyncrasies and therefore, knew (or rather, was able to guess) the origins of a remark. Examine the following phrases the author had extracted from a meeting log. It is highly likely that all these remarks belong to 'felicity':[83]

```
Felicity says 'cant believe that,..'.
Felicity says 'well,...'.
4: well,.. i guess the weight thing is inevidable
18: well,.. this should be dealt with on a personal basis in my opinion.
5: I like to hang with mortals,.. and follow them..
```

Perhaps 'felicity' was not aware of 'her' idiosyncrasy. Perhaps she was not concerned that her remarks might not be anonymous. Perhaps she did not require anonymity. Even if these were true, it did not mean that anonymity was not useful.

The author found that anonymity (or more precisely, the tag technique) had helped him (as the head superuser) to consider the ideas of others without bias or prejudice (ie negative bias). Although the author wanted to give equal treatment to every superuser, it was not always possible. Ideas from certain people were somehow able to command 'more' attention and consideration. The author believes that 'real' equality was only possible when the comments were anonymous.

The pace of a meeting also helped in some way to keep the remarks anonymous. There were too many asays that there was no time available to look for idiosyncrasies or to figure who might have made a particular (anonymous) remark. After reading an asay, the author would usually have enough time to voice his own view before another asay presented itself to be read.

As the months went by, a problem with the tag technique became more apparent. To make an anonymous remark, the author had to remember to use the tag commands. There were numerous occasions where the author forgot. The author had made numerous says when he wanted to make an asay. Such 'accidents' could have serious consequences if one were using an identified username. The concern about using the wrong command was itself part of the problem. The worry was starting to distract the author. The author wanted such stress and accidents to be removed. The `phantom` command was introduced to eliminate solve this problem.

The Anonymous mode

The `phantom` command activates the anonymous mode. Once activated, all communication is automatically anonymised using the tag technique. In other

[83] Of course, the author has no way of knowing for sure. Anyone could have added a comma and ellipsis to implicate 'felicity'.

words, all shouts become ashouts, all tells become atells, and all says become asays—see Transcript 4.6.

```
say let's begin our meeting
You say 'let's begin our meeting'.
phantom
Your character name will be concealed in ALL your conversations.
A [numbered tag] will replace your name for anonymity.
You have been Phantomised!
Type PHANTOM again to dePhantomise yourself.
? say We will have about 10 minutes to raise problems
12: We will have about 10 minutes to raise problems
? say Please remember to use the new phantom command
13: Please remember to use the new phantom command
14: Why are most of the wizard commands logged?
? say So no one will dare to abuse their power?
15: So no one will dare to abuse their power?
16: doesn't drew trust us?
17: are our convos logged too?
```

Transcript 4.6: The anonymous mode ensures that all of one's remarks are (technically) anonymous (Drew's view)[84]

The anonymous mode can be disengaged at any time to make non-anonymous remarks (ie says, tells, or shouts)—see Transcript 4.7. However, it (ie the `phantom` command) would be an additional command that one has to type if one had to make both anonymous and non-anonymous remarks. The anonymous mode is best used if one does not need to make non-anonymous remarks frequently.

```
? phantom
You have been dePhantomised. Your name returns.
say Your conversations will never be logged
You say 'Your conversations will never be logged'.
say Nothing will be logged without your knowledge
You say 'Nothing will be logged without your knowledge'.
say ok, let us continue
You say 'ok, let us continue'.
phantom
Your character name will be concealed in ALL your conversations.
A [numbered tag] will replace your name for anonymity.
You have been Phantomised!
Type PHANTOM again to dePhantomise yourself.
```

Transcript 4.7: The anonymous mode needs to be deactivated to make non-anonymous remarks (Drew's view)

Another problem emerged as more meetings were conducted. Each time the author decided to discuss a matter anonymously, he would have to 'order' the

[84] The ? in the user's prompt serves as an indicator that the anonymous mode is active.

participants (ie superusers) to engage the anonymous mode (or remind them to use asays instead of says). This was not very practical. A way to 'force' everyone in a meeting into the anonymous mode was believed to be necessary.

Mirages (Anonymous rooms)

A *Mirage* is an anonymous room (or a room where the anonymous mode was always active). Remarks made within a Mirage are automatically anonymised by the tag technique. By meeting in a Mirage, the remarks of every participant will be technically anonymous.

The author went a step further. He tried to conceal the presence of the people in a Mirage. The entry and exit messages (eg 'Sam walks in' or 'Sam leaves south') were suppressed. In essence, people could 'sneak' in and out of a Mirage. No one is suppose to know who is or was in a Mirage.

Commands that would reveal the location of users would not work in a Mirage. The `look` command for example, will not reveal who is present—see Transcript 4.8.

```
look
The Shout
This is the gathering place of angry crowds when they feel injustice
has been done since the Judge's chambers are just above.
This place is PHANTOMised. You can't be sure who is around you.
say is there anyone here?
1: is there anyone here?
```

Transcript 4.8: Is there anyone in the Mirage?

To prevent users from revealing their presence or establishing the presence of others, social commands were also disabled—see Transcript 4.9. If 'sam' could be 'hugged', one would know that 'sam' was present.

```
laugh
You can't do that here. You're in a Mirage.
hug sam
You can't do that here. You're in a Mirage.
```

Transcript 4.9: Social commands cannot be used in a Mirage

The decoy checker was also modified to make exceptions for Mirages. A room-level (ie local) checks is not performed in a Mirage. This was done so no one could be sure if there were people in a Mirage. If a local decoy check is performed, one will for example know that there are at least three other users in the Mirage. If there were only three other users online, one would know that they were in the Mirage!

The decoy checker will only perform system-wide (ie global) checks. This means that an asay can be made in a Mirage as long as there are four or more users online. They do not necessarily have to be present in the Mirage. In fact, one would be able to make an asay in a Mirage even if there were no one in it.

To ensure that users understood these technicalities and were not led to make wrong 'conclusions', an extra 'warning' was added—see the text in **bold** in Transcript 4.10.

```
look
The Shout
This is the gathering place of angry crowds when they feel injustice
has been done since the Judge's chambers are just above.
This place is PHANTOMised. You can't be sure who is around you.
You cannot be sure if anyone is here listening to you.
```

Transcript 4.10: A user will be warned that he or she may be the only person in a Mirage

Is it now impossible to determine who is or was in a Mirage? As the author eventually realised, it would be possible for someone in a room *adjoining* the Mirage to know who went into a Mirage. There are several ways of leaving[85] a Mirage but only one for entering—via an adjacent room. How was this problem addressed?

By increasing the adjacent rooms of a Mirage, the number of paths into a Mirage is increased. Clustering several Mirages together also increases the paths.[86] Since there are more than one entry-exit point into a Mirage, it will be impossible for one person to spot everyone entering the Mirage.[87] This was the author's solution.

Since no one should know who is or was in a Mirage, an asay will be traced to everyone on the who-list. If there were a hundred active users online, there would be ninety-nine suspects (even if there were only four people in the Mirage). In a non-Mirage, an asay in a room of four would only lead to three suspects (even if there were ninety-six other active users online). This is one of the benefits of a Mirage.

Unfortunately, no one was as thrilled as the author was. The introduction of Mirages was met with the same lack of enthusiasm as the tag commands and the decoy checker. Perhaps the players and superusers had not found a real need or appreciation for anonymity. Perhaps players wanted to be able to identify one another (to form alliances that would be useful in the game).

Perhaps users did not find anyone in the Mirages to speak to. On a very 'busy' day, there could be as many as ten to twenty players online. Of course, twenty is a small number when compared to 2,652 rooms that made up the Oz

---

[85] The `chamber` and `visit <character>` commands will bring a user from a Mirage to a chamber. The `recall` command will bring the user to the main room. The `jump` command will bring the user to a random room.

[86] A single Mirage would have a maximum of six entry-exit points (ie north, south, east, west, up, and down). A pair Mirages would have a maximum of ten entry-exit points. A cluster of three would create a maximum of fourteen entry-exit points.

[87] A character (ie user) can only be in one room at a time. Unless a person was multiplaying, he or she will only be able to monitor one entry-exit point at a time.

landscape. Even on a busy day, this ratio meant that there was less than a *one percent* probability (ie 20/2652) that a Mirage would be inhabited![88]

How did the author make use of Mirages? The author could not. He could not conduct the superuser meetings in a Mirage because there was no way of knowing whether a non-superuser was present. Furthermore, there was no way of making non-anonymous remarks in a Mirage. What if a Mirage could be locked? What if a private chamber could be converted into a Mirage (and vice-versa)?

Private-Mirages

The `chamber mirage` command converts one's private chamber into a private-Mirage—see Transcript 4.11.

```
look
You see a chamber belonging to Drew.
Isn't it time for a description?
Sue the Cowgirl is here.
Sam the Salmon King is here.
Gates is Microsoft is here.
chamber mirage
Your chamber is transformed into a Mirage.
look
You see a chamber belonging to Drew.
Isn't it time for a description?
This place is PHANTOMised. You can't be sure who is around you.
You cannot be sure if anyone is here listening to you.
say is terminator coming?
15: is terminator coming?
16: he should come
17: terminator is always late
chamber mirage
Your chamber returns to normal.
say I better unlock the doors
You say 'I better unlock the doors'.
chamber lock
*Click* Your chamber is unlocked.
look
You see a chamber belonging to Drew.
Isn't it time for a description?
Sue the Cowgirl is here.
Sam the Salmon King is here.
Gates is Microsoft is here.
```

Transcript 4.11: A chamber can become a private-Mirage (Drew's view)

[88] This is a very simplistic method of calculating probability but the author believed that pin-point accuracy in this case was not critical.

The author found private-Mirages to be extremely useful for conducting superuser meetings. The ability to lock a chamber before it was *miraged* (ie converted into a private-Mirage) meant that security was not compromised. Furthermore, the author could now force anonymity[89] upon the superusers—even upon those that might not want to be anonymous.

How could non-anonymous remarks be made in a private-Mirage? One strategy was to set an agenda for the meeting. Items that required anonymity were identified. When these items were up for discussion, the (locked) chamber was converted to a private-Mirage. Once the item was discussed, the chamber was *de-miraged*.[90] *Miraging* and *de-miraging* would throughout a meeting.

The other strategy was to have a preliminary discussion without anonymity, then proceed to discuss certain issues with anonymity, before closing the meeting without anonymity. In any case, people could always make an anonymous remark (by using an asay) even when the chamber had been de-miraged.

What is the difference between a private-Mirage and those initially implemented? The earlier Mirages (ie *permanent-Mirages*) cannot be de-miraged. With the introduction of private-Mirages, was there still a need for permanent-Mirages? The author believed there was. Once a private-Mirage is de-miraged, the `look` command would reveal who is present in the meeting. Since a permanent-Mirage cannot be de-miraged, no one can be sure who is inside. In other words, a permanent-Mirage will provide users with a greater degree of anonymity than a private-Mirage.

Since the participants in a private-Mirage could be (or were) known, room-level decoy checks were re-introduced. At least four valid parties had to be present in a private-Mirage before an asay was allowed. This ensured that asays made in a private-Mirage would be realistically anonymous.

Minor changes to the numbered remarks

The appearance of the numbered remarks was made to resemble their non-anonymous counterparts. The author wanted anonymity and non-anonymity to coexist seamlessly—see Transcript 4.12 below.

```
asay who thinks this is a better format?
[ 1/you ] say `who thinks this is a better format?'.
[ 2 ] says `i think it is'.
```

[89] A locked private-Mirage would create what the author called confined-anonymity (see page 42).

[90] Before a private-Mirage was de-miraged (ie converted back to an ordinary chamber), Oz would send a warning (reinforced by an audible beep) to everyone in the chamber. The author believed the warning was needed to remind everyone that the remark they were about to would not be anonymous. What was meant to be an asay might have ended as a say had there been no warning.

```
ashout does anyone object this format?
[ 3/you ] shouts `does anyone object this format?'.
[ 4 ] tells [ 3/you ] `i like it'.
atell 4 good! thanks
[ 5/you ] tell [ 4 ] `good! thanks'.
tell 4 great! by the way, who are you?
You tell [ 4 ] `great! by the way, who are you?'.
[ 6 ] tells you `do you really want to know Drew?'.
tell 6 yes
You tell [ 6 ] `yes'.
Sue tells you `just me'.
```

Transcript 4.12: Numbered remarks now have a novel-like appearance (Drew's view)[91]

Other changes include bolding the numbers representing others parties and pairing those representing one's self with the pronoun 'you'.

4.3.2 The second six months of operation

A further six months of observation provided few new conclusions. It was particularly difficult to know how the non-superusers were using the tag technique because Oz did not log the conversations of anyone. Although the author would spend many hours (virtually every single day) making observations, anything that had transpired in private or while the he was not online would have happened without his knowledge.[92]

The activities and conversations of users were not monitored because it was believed to be inappropriate and unethical.[93] Oz had become more than an experiment—it was providing a real service to real people. The author regarded the users as 'customers' rather than subjects in an experiment.

The author did however, observe people using the tag technique for trifle purposes—mainly to 'clown' around. Ashouts ranged from 'guess who is this?' to 'I'm a REDNECK', to more 'playful' remarks like '…is a wimp' or '…is a loser', to somewhat rude remarks that the author has chosen not to describe here.

---

[91] One should note that although 'sue' and 'drew' had revealed their identities to each other, they would still be able to make anonymous remarks. This is because a different number would be used to tag their remarks. The next remark tagged by *7* would not have necessarily come from 'sue'.

[92] All the conclusions described in this chapter are based on what the author had observed in person.

[93] Except for a few instances, all of the transcripts in this thesis are technically *examples* (ie recreations). The events are real (in that something very similar did occur). The transcripts are real (in the sense that they were captured from real working systems). Nevertheless, the transcripts cannot be seen as verbatim accounts. Their main purpose is to serve as examples.

When the numbered remarks were getting tasteless, all that any superuser (including the author) could do was to *caution everyone online or everyone in a particular room*.[94] No one specific could be cautioned. Cautioning did work, however. Perhaps the users believed that the author knew who were responsible. Although guesses could be made, no superuser could have known with absolute certainty who the guilty user was. It was clear that *a more reliable method (than cautions or 'empty threats') was needed to police the use of the tag technique*.[95]

It was also concluded that invisibility was no longer beneficial to anonymity. Not only does invisibility compromise privacy, it can also interrupt the tag technique. There had been occasions where there were sufficient users online but an asay or ashout could not be made because some users were invisible. Since the decoy checker did not (and should not) consider anyone that was invisible, the tag technique could not be used. In fact, it would not be possible to use the tag technique even if there were a hundred users online if they were all invisible!

Why had invisibility not been removed? Every attempt to remove invisibility had been strongly objected by the superusers and players. No one (except the author) seemed to be bothered by invisibility. What function did invisibility serve? The author believed that it simply gave players and superusers a sense of 'power'—the power to spy on people. The author could not see any valid need for invisibility because anonymous remarks could now be made by using the tag commands.[96] No one (not even superusers) should be allowed to be invisible.

The author also concluded that the Oz landscape was far too large. A considerable reduction in the number of rooms should benefit permanent-Mirages (and therefore, anonymity). The size of the landscape should equal the average number of online users (ie approximately ten rooms instead of 2,652). The enormity of the Oz landscape was originally useful for privacy. The (only) way for a group of people to hold a private meeting was to find and meet at a rarely visited or 'faraway'[97] location. With the introduction of

---

[94] Alternatively, the author could have used the `tag <on/off>` command to disable the tag technique completely (ie disable the tag commands, anonymous mode, and mirages).

[95] Even if the superusers were not able to do anything (or were not present), a user could still 'defend' himself or herself. By using the `channel -ashout` command, one would stop receiving (or rather, hearing) ashouts. When earmuffs are put on, one would not receive anything except says and asays. If someone were abusing says or asays, earmuffs would not help. However, one could always move a meeting into one's chamber. After locking one's chamber, any uninvited party could be requested to leave or simply 'kicked out' (using the `chamber remove <character>` command).

[96] Invisibility provided the only way an anonymous remark could be made on the original MERC.

[97] ie a place that could only be reached by successfully traversing a long chain of rooms.

lockable chambers, a vast and complicated landscape was no longer necessary for privacy.

It was no surprise that these plans were met with great objection. Many regular users began to leave Oz. Without their word-of-mouth, new players also stopped arriving.[98] Even superusers began disappearing. Eventually, Oz became devoid of people. Once that happened, it was permanently closed.[99] The last user Oz welcomed was the author:

```
                                  ___
                                 / _ \   _____
                                | / \ | |__  /
                                | \_/ |   / /_
                                 \___/   /____|

                              131.244.8.20 2000

     Oz was created from the Merc (diku mud) code by Kahn, Hatchet, Furey.
          Diku mud originally by Hans Henrik Staerfeldt, Katja Nyboe,
               Tom Madsen, Michael Seifert, and Sebastian Hammer.
                              Ozzy code by Drew.
                                   v 0.8003

 Welcome! By what name do you wish to be known? Drew
```

4.4 Full support for Anonymity

The goal for the next phase of implementation was clear: to ensure that every possible provision or change that would be advantageous to anonymity (and the people requiring anonymity) was made.

4.4.1 The new 'Oz'

All the play elements (including invisibility) were removed. There were no longer any monsters to kill or points to earn. The landscape was reduced to

[98] Although Oz was opened to the Internet, its existence was never made public.

[99] Over the fifteen months of operation, Oz had moved from `131.244.15.53` to `131.244.14.20` to `131.244.200.3` to `131.244.200.4` and finally to `131.244.8.20`. There had been fourteen superusers. Just before Oz was closed, there were two hundred and seventy three active player files (ie two hundred and seventy three different password-protected characters). This figure excludes those characters that belonged to the author. There is no certain way of knowing how many different people had used Oz. Moreover, the figure does not include an unknown number of characters that had been deleted because of inactivity.

thirty rooms.[100] Five out of the thirty rooms were clustered to create a single permanent-Mirage.

The new 'Oz' was called *TwiLIGHT ONE* (T1). T1 was no longer an adventure MUD—it was now a 'serious' conversation system:

```
You have connected to an online interactive communication system.
TwiLIGHT virtual worlds are based on portions of code by Andrew LEE,
Michael Chastain, Michael Quan, Mitchell Tse, Hans Henrik Staerfeldt,
Katja Nyboe, Tom Madsen, Michael Seifert, Sebastian Hammer.
Webpage:  http://www.geocities.com/TimesSquare/7127/
Email:    andrew_lee@bond.edu.au
[privacy]  [anonymity]  [rated G]  [diku-ew2 style cmds]  [australia]

                  -
                  -

                Welcome To
 |   | |     T w i L I G H T    O N E      | |  |   |    |     |      |
           sand.it.bond.edu.au 7777

                  -
                  -

                  -

                  -

Please enter your name:
```

There were other less obvious but nonetheless important changes.

The fundamental policy

The fundamental policy of T1 is to operate without knowing the identities of its users. Any information that could expose the identity of a person was not requested or recorded. This was called the *Limited Information Policy*.

What would be recorded about a user? The new `examine <user>` command reveals everything that is recorded about one's character—see Transcript 4.13.

```
examine terminator
---------------------------  Public Information  ----------------------------
Terminator is back
What do you looking at?
----------  Private Details (may be examined by Senior SuperUsers)  -----------
Terminator is a Resident.
Page pausing (pagelength): 23.   Highlighting (hilite): Off
```

[100] Chambers are not counted as they are dynamically created rooms—ie a user's chamber does not exist until the user is online.

```
Coins: 0
 Character will be purged when coin amount is less than -1.
 Every consecutive 7 days of absence will cost 10 coins.
------------------------------------------------------------------------------
```

Transcript 4.13: Checking what is recorded about one's self (Terminator's view)

Every piece of information on record is classified as either public or private. The `examine <user>` command applied to another user will only return the information in the public portion—see Transcript 4.14. Anything a user uses to describe himself or herself (eg the username ie 'drew', title ie 'keeps everything in order', and description ie 'You see someone with very sleepy eyes.') is considered public information. Everything else is deemed private.

```
look
The TwiLIGHT square
[Exits: north east west up]
You are standing in the middle of a very large open square tiled with marble.
Drew keeps everything in order is here.
examine drew
----------------------------  Public Information  -----------------------------
Drew keeps everything in order
You see someone with very sleepy eyes.
------------------------------------------------------------------------------
```

Transcript 4.14: Learning more about 'drew' (Terminator's view)

The description field provides a way to control the degree of one's anonymity. If 'drew' wanted to reveal more about himself, he could replace 'You see someone with very sleepy eyes' with more useful information—see Transcript 4.15.

```
edit description
Type your personal description. When done, type .END by itself on a new line.
Drew is Andrew Lee in real-life.
You can e-mail me at: andrew_lee@bond.edu.au
Write me at: School of Information Technology, Bond Uni, Australia 4229
Phone me at: 61-75-55953380
.end
OK.
examine drew
----------------------------  Public Information  -----------------------------
Drew keeps everything in order
Drew is Andrew Lee in real-life.
You can e-mail me at: andrew_lee@bond.edu.au
Write me at: School of Information Technology, Bond Uni, Australia 4229
Phone me at: 61-75-55953380
----------  Private Details (may be examined by Senior SuperUsers)  -----------
Drew is the Head Superuser.
Page pausing (pagelength): 23.   Highlighting (hilite): Off
Coins: 0
No Purge ON.
------------------------------------------------------------------------------
```

Transcript 4.15: The description field can be used to identify one's self (Drew's view)

The private section would only be accessible to the user and to a senior (ie 'veteran') superuser. Was it necessary for any of the information in the private section to be available to anyone apart from the user? Although there were no concrete reasons, the author believed it was prudent to include such a provision. If T1 were a commercial service, this provision would have allowed a user's account details (eg the user's account number, amount owning, or credit card details) to be examined by the administration (and user).

Penalising Anonymous users

To satisfy the limited information policy, the banning procedure inherited from Oz (or more precisely, from MERC) was also changed. Previously, the head superuser would use the `users` command to extract a guilty party's IP address. The `ban` command would then be used to impose a ban of the IP address—see Transcript 4.16.

```
users
[  1  0] Drew@SAND.KOWANDE.Bond.edu.au
[  2  0] Tester@SURF.KOWANDE.Bond.edu.au
2 users
ban surf.kowande.bond.edu.au
OK.
ban bond.edu.au
OK.
```

Transcript 4.16: The process of banning a computer and site on Oz

If the guilty party had left the system (or had been disconnected by a lower-ranking superuser),[101] the head superuser would have to obtain the IP address from the *login logs*.[102] The logs would list which user had logged on from where and when.

Automating the banishment procedure

A different procedure was used on T1. The new `autoban <user>` command replaced the `users` command. A superuser would now only need to specify whom to expel. T1 would obtain the guilty user's IP address from the network protocol and record it in the *banned-IP file*. T1 would then disconnect the guilty user. The benefit of automating the banishment procedure is that no one will need to know anyone's IP address.

[101] The `ban` (and `users`) command was only available to the head superuser (ie the author). Other superusers could only: prevent a user from doing anything (using the `freeze <user>` command), or remove the user from the system (using the `disconnect <user>` command).

[102] This could only be done from the operating system.

What if an 'innocent' user was affected by an IP address ban? How could an IP (address) ban be reversed? T1 was designed to lift an IP ban after thirty days. Should an innocent party be affected, he or she would have wait the thirty days.[103] An IP ban could be lifted before the thirty-day period but it would require the author to edit the banned-IP file (from the operating system).[104]

Would the limited information policy be violated because the head superuser could examine the banned-IP file? 'No,' since anyone found abusing anonymity would have lost his or her right to anonymity. Every pair of username and IP address recorded in the banned-IP file would be a fact about a guilty user. Although the IP address of an 'innocent' user might be present in the banned-IP file, it would not be paired with the innocent user's username. In other words, the administration (or more precisely, the author) should not know the IP address of an innocent user.

To 'reduce' the likelihood of IP bans affecting innocent users, the `autoban` command was deliberately designed to ban specific IP addresses (ie a machine) and not entire sites.[105] In fact, there were other types of bans would be used before resorting to the `autoban` command. The `sulock` command allows T1 to be temporarily closed to non-superusers. The `vislock` command allows the system to be temporarily closed to visitors (ie users without a password-protected username).

While testing the `autoban` command, the author stumbled across a potential problem. If the guilty user were online, T1 would be able to obtain the user's IP address from the network protocol. Since IP addresses were no longer recorded in the login logs (to comply with the Limited Information Policy), how would T1 ban someone who was not online? A crafty troublemaker might have waited until there were no superusers online before creating any trouble and left the system before the author (ie head superuser) could take any action.

Temporary logging of IP addresses

The solution was to record the IP address of every user in a *temporary login log*. The temporary login logs would only contain pairs of usernames and IP addresses.[106] Two temporary login logs will exist—each holding twelve hours

[103] Should a superuser be affected by an IP address ban, he or she would know about the secret login method (ie the 'backdoor'). The backdoor would only accept login attempts from superusers (ie superuser characters).

[104] The ability to retrieve a violator's IP address from the banned-IP file also enables the author to convert a thirty-day IP address ban to an indefinite IP address ban or a site ban (using the `ban <IP or site address>` command). By knowing a violator's IP address, an e-mail complaint can also be sent to the violator's Internet Service Provider.

[105] The `ban <site>` command was not removed to enable a site ban to be erected if necessary.

[106] The login logs contained usernames, dates, login times, and logout times.

worth of logins. When a third temporary login log is needed, the older of the existing two will be automatically deleted. Although the IP addresses of 'innocent' users will be kept, they will only be held for a maximum of twenty-four hours.

When the `autoban <username>` command is invoked, T1 will first check to whether the user is online. If the user were no longer online, T1 would scan the temporary login logs. It would look for the last user with the matching username, retrieve the IP address, ban the address, and keep a record of the address in the banned-IP file.

To ensure users understood that their IP addresses were temporarily logged, a warning (see the text in bold) was added to the login screen:

```
Warning: Each time you use this system, your network address is held for 24 hours
in discreet. If you had observed the conditions of use, your network address
would be automatically forgotten after 24 hours. If you had not, it would be used
to keep you off this system.
Please enter your name:
```

In addition, the contents of the temporary login logs are encrypted. It will not be possible for anyone to extract any information from those logs. The key used to encrypt the temporary login logs will be randomly generated by T1. Should T1 'crash' or be shut off, the contents of the temporary logs would become useless—not even T1 would know how to decrypt the data it had encrypted.

Enhancements to the Alias and Pretend Techniques

When relying solely on the alias or pretend technique, username changes are necessary for one to switch between semi-anonymous (or semi-identified) and identified remarks. To facilitate username changes, the `morph` command was added—see Transcript 4.17 below.

```
Luke says `I think Star Trek is dumb'.
Luke shouts `I think Star Trek is dumb'.
Gates laughs at Luke mercilessly.
Luke slaps Gates.
Vader slaps Gates.
roll vader
You roll your eyes at Vader.
Gates rolls his eyes at Vader.
shout trekies, visit luke
You shout `trekies, visit luke'.
Sue comes in for a visit.
morph
Are you sure about morphing (Y/N)? y
New name to morph to: picard
   This character name already exists.
   Unauthorised access will not be tolerated.

   To choose another name, simply press [return] at the password.
```

```
Password:
Enter another name: borg
Your character name is borg. Is that right (Y/N)? y
Is this character a Male, Female or Not applicable (M/F/N)? n
The TwiLIGHT square
[Exits: north east west up]
You are standing in the middle of a very large open square tiled with marble.
visit luke
[Exits: none]
You see a chamber belonging to Luke.
Isn't it time for a description?
Luke Skywalker is here.
Vader is here.
Gates is Microsoft is here.
Sue the Cowgirl is here.
Kirk James T is here.
Ally Cat is here.
Bob is JR's brother! is here.
Capt Picard of the USS Enterprise is here.
Luke says `hello borg!'.
say WE ARE BORG...RESISTENCE IS FUTILE!
You say `WE ARE BORG...RESISTENCE IS FUTILE!'.
```

Transcript 4.17: The user 'terminator' *morphed* into 'borg' (Terminator's view)

The changeover will be faster because 'terminator' does not have to quit the system and 'borg' does not be forced to read the usual welcome messages or the terms of usage. The changeover will be less noticeable because the character 'terminator' is removed from the system without leaving the usual 'Terminator has left T1' message. No one should realise that 'terminator' has left (unless someone typed the `look` command or tried to send 'terminator' a private remark).

The user 'borg' (ie previously 'terminator') will however, enter the system in the usual manner. Concerns for privacy dictated that no one be allowed to 'sneak' into a non-Mirage. In other words, the user 'borg' has to find 'his' way back to where he was (ie back to the room where the debate was happening).

Enhancements to the Tag technique

The `aemote` and `athink` tag commands were added to allow users to be more 'expressive'—see Transcript 4.18 below. The slash and the blank spaces within the square brackets were also removed to simplify the appearance and reduce the length of a numbered remark.

```
[8] says `will microsoft still be around in 2000?'.
Gates thinks o O ( WHAT KIND OF QUESTION IS THAT? )
Gates thinks o O ( Of course we will! )
[9] giggles.
aemote laughs loudly
[10 you] laughs loudly.
[11] thinks o O ( what's so funny? )
```

```
athink What could possibly happen?
[12 you] think o O ( what could possibly happen? )
[13] thinks o O ( doesn't 12 know murphy's law? )
Gates say `Who's Murphy? Is this person under my employ???'
amote can't stop laughing
[14 you] can't stop laughing
```

Transcript 4.18: New tag commands to enhance interaction

The other significant enhancement was the decapitalisation of the numbered remarks—see Transcript 4.19. This was done to remove any capitalisation idiosyncrasies. A built-in dictionary and thesaurus to standardise spelling and choice of words would have been beneficial but were not implemented because of the author's lack of expertise and the fact that off-the-shelf products such as Casady & Greene's *Spell Catcher* or Linguisoft's *Grammarian* were available.[107]

```
asay Who AM i?
[1 you] say `who am i?'.
asay HEY! Why is EveryThing In Lower CaseS?
[2 you] say `hey! why is everything in lower cases?'.
asay I LiKe to be DiFFerenT!
[3 you] say `i like to be different!'.
say I like to be differenT!
You say `I like to be differenT!'.
[4] tells you `your remarks won't be as anonymous as they could if you were
different'.
```

Transcript 4.19: The *capitalisation checker* helps to remove capitalisation idiosyncrasies

Another way of addressing idiosyncrasies would be to request someone to be one's messenger—see Transcript 4.20. By asking 'bishop' to convey one's remarks, one's idiosyncrasies become intertwined with the messenger's.[108]

```
Bob tells you `what do you think about macs?'.
atell bishop please tell bob that macs are really great computers and are really
great to use. Thanks!
[5 you] tell Bishop `please tell bob that macs are really great computers and are
really great to use. thanks!'.
Bishop tells [5 you] `Macs are great and great to use?'.
atell bishop yes, please tell that to bob
[6 you] tell Bishop `yes, please tell that to bob'.
```

Transcript 4.20: Using a messenger to hide one's idiosyncrasies

[107] Alternatively, speech-to-text technologies such as Apple Computer's *PlainTalk* or Dragon System's *VoicePower Pro* could be used to create sentences with words spelled (flawlessly).

[108] Of course, this assumes that 'bishop' did not just 'copy and paste' one's remarks to 'bob'.

The feedback of the decoy checker was also changed—see Transcript 4.21. In addition to protecting a user's authorship anonymity, the decoy checker would now be helping to educate the user.

```
asay this is a test
Sorry, can't do that right here. Too few people to be anonymous this way.
If not, they would have read:  [1] says `this is a test'.
```

Transcript 4.21: Decoy checker giving feedback that is more useful

4.4.2 The McTwilight Telnet client

The McTwilight (MT) Telnet client (see Figure 4.1) was created to support two other provisions.

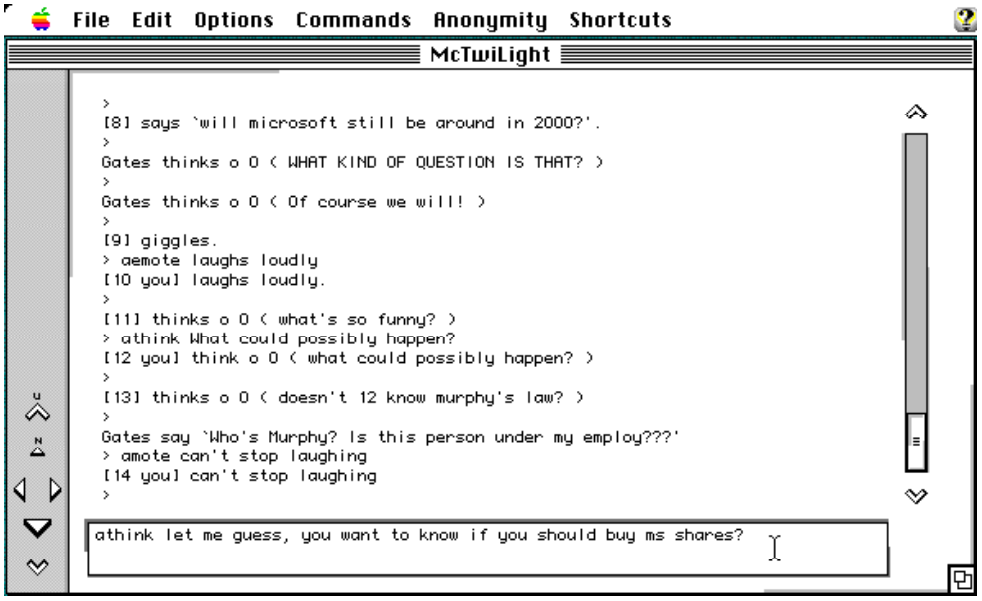

Figure 4.1: The Macintosh-based *McTwilight* Telnet client

Keyword screening

To help screen certain keywords out of a user's remarks, MT supports a feature called the *word-blocker*. The work-blocker will look for keywords (specified by the user) in every remark a user makes. When a match is found, MT will allow the user to make a correction—see Figure 4.2 on the next page.

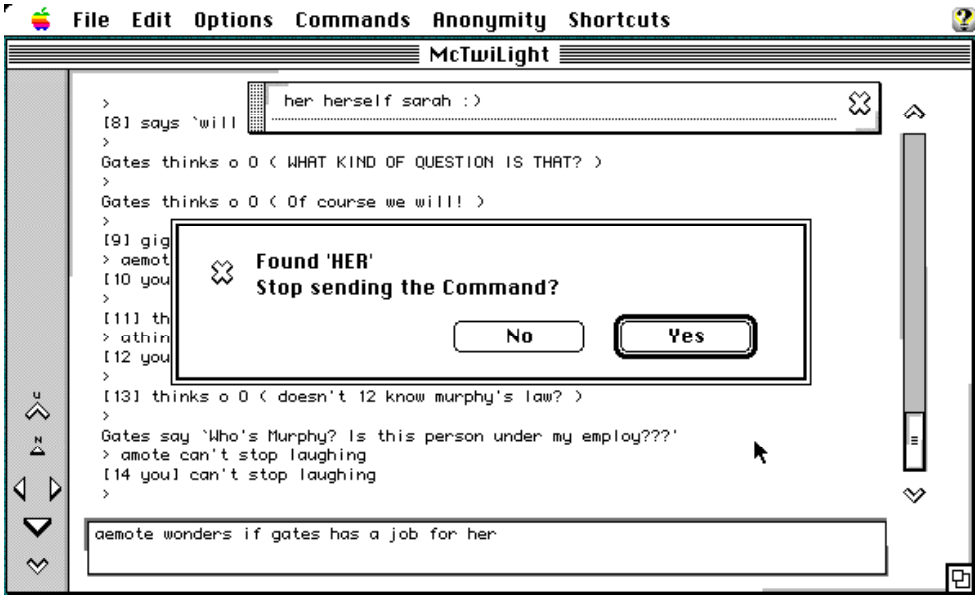

Figure 4.2: The word-blocker has detected an 'illegal' keyword

The word-blocker provision does not put any strain on the T1 server because the MT client will be performing the screening. The list of keywords (which can contain sensitive information) is managed by the MT client—ensuring that T1 server (and service) can continue to comply with the Limited Information Policy.

Message encryption

Although T1 has been designed to protect a user's privacy, intrusions on privacy could still occur (at the packet level). Transmissions over a network can be intercepted (as it passes through various computers) and analysed. Furthermore, transmissions of a textual nature are particularly easy to analyse. Encryption will provide an additional level of privacy. Even if one's messages were intercepted, no one should be able to understand what was communicated.

To create encrypted asays and atells, one simply makes the appropriate settings (see Figure 4.3 on the next page) and engage the anonymous mode (via the `phantom` command). MT will automatically encrypt any says (and tells) while T1 will automatically convert the encrypted says (and tells) into asays (and atells).

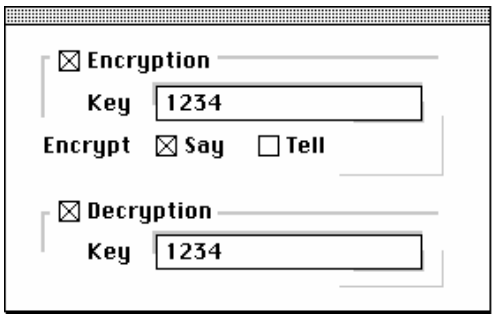

Figure 4.3: Encryption and Decryption options in MT

If a group of people had prearranged to use a common encryption key, the group would be able to decrypt (and hence, understand) each other's remarks. Without the right decryption key however, an encrypted remark would appear as garbles:

```
Gates says '~Lfh@llcme ak i2 hou ay r@yoX8<ilI ob8n @rsm k`'_c2sa?~'.
```

As with screening, encryption and decryption is handled by the MT client and not the T1 server. The server will not even know that a message is encrypted.

4.4.3 The T1 experience

Unlike Oz, the existence of T1 was made public. T1 was listed on the *MUD Connector,*[109] announced on various Internet newsgroups, and made the default destination on every copy of MT distributed.[110]

T1 was open for a one hundred and twenty-six days.[111] During that time, a total of 837 different usernames were recorded, of which forty-four were reserved (ie password-protected).[112]

There were very few observations to make since people did not stay very long. Over half the visits (approximately 55%) lasted less than ten minutes.[113] In fact,

[109] `http://www.mudconnect.com/`

[110] MT was distributed through the non-commercial software distribution channels such as the *INFO-MAC* (`ftp://wuarchive.wustl.edu/systems/mac/info-mac/`) and *UMICH* (`http://www.umich.edu/~archive/mac/`) archives.

[111] T1 was abruptly forced to close because the host used to run T1 was no longer accessible by the author.

[112] These numbers do not include those characters belonging to the author. Again, there is no precise way of knowing the number of people using T1. However, since people were warned not to reserve multiple usernames, one could assume that there were at least forty-four different people.

[113] This data was gathered from 1,472 login-logout records. This figure excludes the data generated by the author or the superusers. There were four superusers on T1. If they were considered, the results would have been skewed since the author and superusers would have undoubtedly stayed online for longer.

the median was only six minutes and forty-two seconds. Chart 4.1 provides a summary of the data.

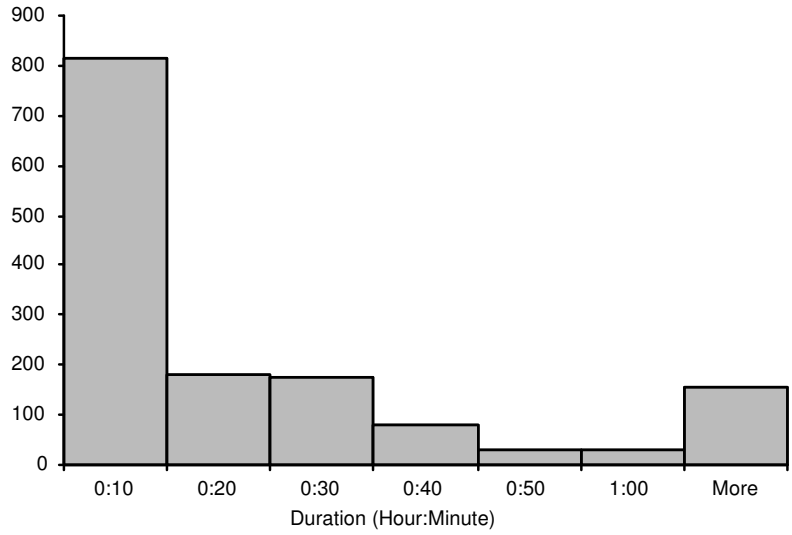

Chart 4.1: Histogram showing most of the visits lasted less than ten minutes

Why did so many people stay so briefly? Unlike Oz, there is nothing for a single user to do on T1. There was an average of eleven logins a day. In other words, approximately one person visited T1 every two hours. Most people would have rightly logged off when they found no one else online.

The main room of Oz was almost always inhabited. It was the place where friends got together and strangers became friends. In contrast, the main room of T1 was almost always deserted (except for when a superuser was on duty). Where were the regular users? Whether alone or in pairs, regular users usually remain in chambers. Most of the time, the author found that their earmuffs were enabled and chambers locked. The author had the impression that the regular users were people that had made prior arrangements to meet on T1. Otherwise, they would have spent their time at the main room for an opportunity to meet new people. Perhaps visitors did not stay long because no one wanted to socialise with them.

The low number of online users could also explain why so many usernames were not protected. Unless there are four or more active participants in a conversation, the decoy checker will not permit anyone to employ the tag technique. Perhaps people were forced to resort to the alias technique (ie 'one-off' usernames) to be anonymous.

Perhaps there were too many established online services on the Internet. While T1 was usually 'empty', places like Foothills had large numbers of people. Perhaps people did not find anything 'special' about T1. Perhaps people chose not to use T1 until they had a real need for conversational anonymity. Perhaps more time was needed for T1 to establish itself.

4.4.4 'Improving' upon T1

When the author secured use of a new host, he continued to search for ways to benefit anonymity and anonymous users. Putting anonymity above other considerations resulted in T1. *What if there was no other consideration besides anonymity?* This was the goal of the third round of transformation.

One of the things that should be beneficial to authorship anonymity is the removal of all global participant lists. Without a who-list, one will not know who may be (or may have been) online. Without a who-list, one will not be able to trace an ashout to a list of users. The `who` command was modified to report the number of people online instead of whom they were—see Transcript 4.22. The absence of a global participant list should not change the interaction between people in a room. Everyone will know who is present because the `look` command was retained.

```
who
There are currently 5 persons and 1 officer online.
```

Transcript 4.22: A summary replaces the who-list

The new `mark` command allows users to change their point of entry (from the default arrival room). Any room except a chamber could now be used to enter the system—see Transcript 4.23. This command will effectively allow users to 'sneak' onto the system.

```
look
Captain's Quarters
[Exits: down]
mark
You mark this spot. From now on, you will enter the station here.
```

Transcript 4.23: Marking an alternative entry point

To increase the choice of entry points, the author tripled the landscape to sixty rooms. Would not increasing the number of rooms be disadvantageous to permanent-Mirages? 'Yes,' but the author considered the problem to be negligible—sixty rooms were still less than 2,652 (which was the number of rooms on Oz).

The new 'T1' was renamed *TwiLIGHT Two* (T2):

```
You have reached a public interactive conversation system.
The system is based on portions of code by Andrew Lee, Michael Chastain,
Michael Quan, Mitchell Tse, Hans Henrik Staerfeldt, Katja Nyboe, Tom Madsen,
Michael Seifert & Sebastian Hammer.

MUD style commands | Anonymity                  ./\.
                                              ./   .\.
Welcome To                                   ./       :\.
T w i L I G H T    T W O                    ./    .. ..:::\.
The Space Station                        ./               :\.
serpent.dstc.bond.edu.au 7777         ./                .:::\.
                                    ./             .....::::::\.
                                  ./                      ...::::\.
                                /....       ... ......::::::::::::\
                                    /.....             :::::::\
Warning: Each time you use this           \          ..../
system, your network address is            l.....  .::::l
held for 24 hours in discreet.                ./  ..\.
If you had observed the conditions of use,     \   ./
your network address would be automatically     \ ./
forgotten after 24 hours. If you had not,        \/
it would be used to keep you off this system.

Please identify yourself:
```

If one were to enter the 'space station' at a permanent-Mirage (and did not leave the Mirage), no one should know that one was online. Since the `look` command will be disabled in a Mirage and one's remarks will be tagged by numbers, one's username does not appear anywhere. If one's remarks were free from idiosyncrasies, one's presence would have been completely concealed—potentially creating absolute-anonymity.[114]

A cluster of five rooms was designated as the station's only permanent-Mirage. The rooms in the permanent-Mirage are connected to each other but not to the rest of the station—stopping people being spotted 'walking' in or out of a permanent-Mirage. The `recall`, `leave`, or `jump` command will need to be used to leave the permanent-Mirage.[115] The new `Mirage` command will bring a user into one of the five rooms in the cluster of permanent-Mirages—see Transcript 4.24 on the next page.

[114] Absolute-anonymity was explained on page 41.

[115] The `jump` command brings a user to a random public room on the station. The `recall` and `leave` commands will bring a user back to the main room (ie the *Station Square*).

```
look
Station Square
[Exits: north east south west down]
Picard is here.
Luke is here.
The transporter beams Picard somewhere.
mirage
The Cone
[Exits: north east south west] [Mirage]
It is too dark here. You can't be sure who is around you.
```

Transcript 4.24: The **mirage** command in action

Transcript 4.24 also shows what one will see when a user (ie 'picard') goes into the permanent-Mirage. Would not the words 'The transporter beams Picard somewhere' expose the fact that 'picard' went to the permanent-Mirage? 'Not necessarily,' because the same feedback is used in other commands (ie the `chamber`, `visit`, and `jump` commands). This is intentional so no one will know for certain where another user went.

T2 was now complete. The author could not find any other changes that would be beneficial to conversational anonymity. Although the option of opening T2 to the public was available, it was not taken. It was believed that T2 and T1 were fundamentally identical. In fact, most of the key provisions in T2 (eg the tag commands) were already present in Oz. The author believed that observations would consume too much time. Furthermore, observations would not show that T2 was the most ideal environment for conversational anonymity. The next chapter describes several new methods of analysis and their results.

Chapter 5

Analysis and Discussion

5.1 Laboratory experiments

Eight university students of different ethnicity, proficiency in English, and level of acquaintance[116] were asked to participate in a series of experiments. A special four-room version of *TwiLIGHT Two* (T2) was created for this purpose.[117] The eight students were distributed across a laboratory of twenty computers.

Session 1: Control 1

In the first session, everyone was free to choose his or her username. Everyone was taught how to use the non-anonymous communication commands (ie say, shout, and tell) and social commands. At no point was the objective of the experiment revealed. No one was instructed to conceal his or her identity.

Some students were complementing their online conversations with verbal communication (by shouting across the lab). In the end, everyone was laughing, shouting, and typing. By following the online and verbal conversations, the author was able to discern which username belonged to whom. Two participants were in fact, using their actual names as their username. Some freely revealed their identity when asked. Some even revealed who other users were.

Once the usernames became identified, the author was able to record idiosyncrasies. The author jotted down his observations. After twenty minutes, the participants were asked to exit the system. Each student then stood up to reveal his or her username.

[116] Two of the subjects were siblings.

[117] A smaller landscape was used to prevent the students from wasting time 'wandering' around.

Session 2: Control 2

In the second session, the students were specifically instructed to choose an anonymous username and to protect their identity as best they could. Their objective was to determine the person (ie identity) behind every username. No one was allowed to make any verbal comments once the experiment began. There was to be absolute silence. Furthermore, the laboratory lights were turned off during the experiment.

Except for one person, everyone was recognised by at least one other person.[118] It was not certain how people had managed to recognise one another. Personally, the author was able to identify several usernames because of the idiosyncrasies observed from the previous session. Once the author knew the idiosyncrasies of a particular person, the author could recognise the person even though a different username was used.

It was a surprise however, that one person was able to escape exposure. It was agreed that this person's *online personality was distinctly differently from his real-life demeanour.* This person had successfully masked his real-life identity by changing his personality.

Session 3: The Test

For the third session, students were introduced to the tag technique and anonymous mode. They were asked to select a new anonymous username and have the anonymous mode engaged. The objective of the experiment remained the same.

The students had twenty minutes to determine who each other was. The results showed that there was only one user that was correctly identified. More remarkably, everyone was able to expose this person. Had the students been colluding? There was a more rational explanation.

One particular student had developed an obsession using the `whip <user>` social command.[119] This obsession began in the first session and continued to the third. Although social commands were not been allowed in an anonymous room (ie Mirage), they were not prevented when the anonymous mode was engaged—an oversight by the author.[120] Since the tag technique did not anonymise the 'output' of social commands, everyone was able to trace the whipping habit to a username. Since everyone knew which person had the habit of whipping, they were able to tie the username to the person—resulting in one common exposed username.

---

[118] The author's personal 'speculations' were not included in the results.

[119] The `whip` command produces the following comical message: `<username> whips you in the rear with a wet towel *POW!* That hurts!`

[120] Social commands were later disabled during the anonymous mode.

Analysis of the experiments

Table 5.1 describes the results of the laboratory experiments. The figures appeared to suggest that the students were *more protected in the third session than the second*. The use of the alias and tag techniques appeared to have provided a 75% improvement in protection than the use of the alias technique alone.

Table 5.1: Results of the Laboratory Experiments

	Number of Exposed Usernames	*Percentage of Exposure*
Session 1	N/A[121]	N/A
Session 2	7	88%
Session 3	1	13%

The *Anti-Anonymity Checklist 1* (AAC1) analysis also confirmed that students were most protected during the third session—see Table 5.2.

Table 5.2: AAC1 Analysis of the Laboratory Experiments (continued next page)

	Factors	*1*	*2*	*3*
1	Password-protected username is supported.	2/2	2/2	2/2
2	Personal details submitted? No.	0/2	0/2	0/2
3	Admin has access to personal details? No.	0/2	0/2	0/2
4	Compromised privacy? No.	0/2	0/2	0/2
5	Lack of Confidentiality? No.[122]	0/2	0/2	0/2
6	Remarks tagged by username? Yes, except for Session 3.[123]	1/1	1/1	0/1
7	Deanonymisation of anonymous remarks? No.	0/1	0/1	0/1
8	Known Idiosyncrasies? Idiosyncrasies began to emerge in Session 1	0.5/1	1/1	1/1
9	Private communication supported? Yes.	1/1	1/1	1/1
10	Meetings can be restricted. Closed meeting? Yes.	2/2	2/2	2/2

[121] There is no data for Session 1 because users had not been asked to expose each other's username.

[122] Although users knew one another's identity, it was not due to lack of confidentiality by the administration.

[123] In Session 3, remarks were tagged by numbers.

	Factors	*1*	*2*	*3*
11	Proximity? Users were in the same laboratory.	2/2	2/2	2/2
12	User is not anonymous to some participants. In Session 1 users were not told to conceal their identity.	2/2	0/2	0/2
	AAC1 Rating	10.5/20	9/20	8/20
	AAC1 Percentage	53%	45%	40%

Although the trends in Table 5.1 and Table 5.2 were the same, the magnitude of change was different. The percentage of usernames exposed dropped by 75% (from Session 2 to 3) while the AAC1 percentages only dropped by 5%. One explanation for this is that the AAC1 percentages represented 'predictions' while the percentages from counting (exposed usernames) represented actual data—predictions are not always actualised.

A more accurate explanation is perhaps that a simple count of exposed usernames does not measure all the loss of anonymity that could have occurred. Some anonymity is lost:

1 when a username is no longer mysterious,
2 when a person's remark can be recognised (because of idiosyncrasies in the remark), and
3 when a person's real name is known.

Counting the number of exposed usernames merely considers one of the three dimensions.

Even when the author was able to trace a remark to a person (ie expose a remark), he could not trace the exposed remark to a username. It was as though users did not have any username. If the exposed remarks were tagged by usernames, the author would have been able to expose several usernames. The improvements in Session 3 should be more modest than the 75% (suggested by counting exposed usernames). Perhaps an improvement of 5% (as indicated by the AAC1 analysis) would be more accurate.

Does a 5% improvement suggest that the tag technique was of little additional benefit? The author does not think so. The tag technique had made the task of profiling people more difficult. Although the author noticed new idiosyncrasies in Session 3, he did not know whose idiosyncrasies they were. Had the tag technique been used from Session 1, the author might not have been able to profile every student.

The author strongly believes that *the benefit of the tag technique (or any other provision for anonymity) should not be measured quantitatively*. What is the price for anonymity? Someone that thinks anonymity is invaluable will also find every additional protection invaluable.

If the anonymity of the users in Sessions 1 and 2 was the type occurring naturally,[124] the average of Sessions 1 and 2 (ie 49%) could be taken to represent MERC and any other conversation system that simply 'tolerates' anonymity. If that were accepted, is T2 approximately 9% (ie 49-40%) more 'superior' than MERC? Since MERC does not support the tag technique, it is perhaps true to say that T2 is more 'superior'. However, the 9% 'advantage' is a figure derived from three AAC1 percentages representing a specific set of circumstances—ie an online meeting between a group of people in a computer laboratory. Other circumstances might see a greater or lesser degree of 'advantage'. In short, the 9% advantage should not be treated as a valid comparison. The AAC1 analysis should not be used to make general comparisons between systems.

The PANIC model and notation

The author developed the PANIC model and notation based on what he understood about loss of anonymity. The acronym *PANIC* was derived from key elements that would contribute to loss of anonymity: *P* for a *person* (or a 'face'), *A* for an *alias* (or a username), *C* for a *creation* (or in this context, a remark), *I* for an *idiosyncrasy* (ie a clue), and *N* for a *name* (or more specifically, a full real name or real-life name). The PANIC elements were then used to construct simple 'equations' that described how anonymity could be lost (see Table 5.3). Subscripts were added to give the elements value. For example, to refer to person Sarah Parker, the author would use the expression $P_{\text{Sarah Parker}}$. To refer to the name 'Sarah Parker', one could use the expression $N_{\text{Sarah Parker}}$. The subscript '?' was used to show anonymity—for example, $P_?$ represents an anonymous person.

Table 5.3: Loss of Anonymity in PANIC (continued next page)

English Expression	*PANIC Expression*
An anonymous person becomes non-anonymous once the person's real name is known.	$P_? + N_x \Rightarrow P_x$
For example, the name *Sarah Parker* ($N_{\text{Sarah Parker}}$) *associated to*[125] an anonymous person ($P_?$) makes the person non-anonymous (ie	$P_? + N_{\text{Sarah Parker}} \Rightarrow P_{\text{Sarah Parker}}$
An anonymous username or[126] remark associated to (or traced to) a non-anonymous person becomes a non-anonymous username or remark.	$(A\|C)_? + (P\|N)_x \Rightarrow (A\|C)_x$

[124] ie created because of non-face-to-face interaction and use of typed remarks.

[125] The operator '+' is used to indicate that there is a *proven (or verifiable) connection* between two elements,

[126] The operator '|' is used to replace the word 'or'.

English Expression	*PANIC Expression*
An anonymous person or remark connected to an identified alias (ie non-anonymous username) becomes non-anonymous (or non-mysterious). In other words, a person using a non-anonymous alias is not anonymous.	$(P\|C)_? + A_x \Rightarrow (P\|C)_x$
An idiosyncrasy observed in[127] a person's message exposes (or becomes) the person's idiosyncrasy. Equally, an idiosyncrasy associated to a known person or identified username causes the idiosyncrasy to be identified.	$C_x(I_?) \Rightarrow I_x$ $I_? + (P\|A)_x \Rightarrow I_x$
A known (or exposed) idiosyncrasy makes a remark non-anonymous.	$C_? + I_x \Rightarrow C_x$
A message tagged by (ie associated to) an identified username renders the remark non-anonymous.	$C_? + A_x \Rightarrow C(A_x) \Rightarrow C_x$
A person's idiosyncrasies or alias exposes the presence of the person.	$P_? + (A\|I)_x \Rightarrow P_x$

It was also discovered that most other concepts relating to anonymity could also be described in PANIC—see Table 5.4.

Table 5.4: Other concepts described in PANIC (continued next page)

English Expression	*PANIC Expression*
The use of a false name or idiosyncrasy to conceal (or replace)[128] one's real name and idiosyncrasy creates *Hidden-Anonymity* (HA). When X replaced his or her real-life name and idiosyncrasies with that of another person (ie Y), X 'became' another person (and received Hidden-Anonymity).	$P_x - (N \,\&\, I)_x + (N\|I)_y \Rightarrow P_y \,\|\, P_{?HA}$
Protected-Anonymity (PtA) is created when one person agrees to conceal another person's real name. When X and Y agree to conceal the real-life name of X, Y receives Protected-Anonymity.	$(P_y \,\&\, P_x) - N_x \Rightarrow P_y \,\&\, P_{?PtA}$

[127] The expression 'C(I)' is used to show that 'I' is observed in 'C'.

[128] The operator '-' is used to indicate removal (or concealment).

English Expression	*PANIC Expression*
A user who manages to conceal his or her real name receives *Profile-Anonymity* (PfA).	A_x - $N_x \Rightarrow P_{?PfA}$
Confined-Anonymity (CA) is created when a user is able to conceal his or her remarks.	$C(A_x)$ - $A_x \Rightarrow C_{?CA}$ C_x - $A_x \Rightarrow C_{?CA}$
CA is created when a remark is traced to a group of people.	$C_?$ + $(P\|A\|N)_{x,y,z} \Rightarrow C_{?CA}$
CA is created when the username of an identified person is not known.	P_x - $A_x \Rightarrow A_{?CA}$
CA is also created when a username is traced to a group of people.	$A_?$ + $P_{x,y,z} \Rightarrow A_{?CA}$
Absolute-Anonymity (AA) occurs when one is able to remove all the evidence connected to one's remark.	$C(P\|A\|N\|I)_x$ - $(P \& A \& N \& I)_x \Rightarrow C_{?AA}$ In other words, C_x - $(P \& A \& N \& I)_x \Rightarrow C_{?AA}$
The *Alias Technique* works by associating a remark to anonymous alias instead of a person or identified username.	$C(P\|A\|N)_x$ - $(P \& A \& N)_x$ + $A_? \Rightarrow C_?$ C_x - $(P \& A \& N)_x$ + $A_? \Rightarrow C_?$
The *Tag Technique* tries to sever the association between a remark and a user (or person).	$C(P\|A\|N)_x$ - $(P \& A \& N)_x \Rightarrow C_?$ C_x - $(P \& A \& N)_x \Rightarrow C_?$

Besides providing a way to describe anonymity-related events, the PANIC expressions (or 'equations') has also provided the author with further insights into anonymity. The expressions seem to show that the scope and potential for loss of anonymity increases with the inclusion of more PANIC elements. It would seem that *the five PANIC elements need to be kept apart or removed from a conversation to reduce loss of anonymity.* This 'discovery' has unknowingly been the author's strategy from the onset. In essence, the limited information policy (see page 75) attempts to remove real names (ie the *N* element) from the expression. The tag technique attempts to separate usernames (ie the *A* element) from remarks (ie the *C* element). Mirages attempt to remove usernames altogether. The capitalisation checker attempts to remove capitalisation idiosyncrasies from remarks (ie remove some of the *I* element).

5.2 Pseudo-scenario analysis

The Longman dictionary defines the word *scenario* as 'a written description of a possible course of action or events.' In Lee 94, the author used scenarios to describe various 'imaginary' systems in action. A *pseudo-scenario* describes a

'real' system in action. It describes a factual course of action or events. Although the circumstances might be 'invented', the events that follow are based on what a system allows. The transcripts found throughout Chapters 2, 3, and 4 can be considered pseudo-scenarios.

Pseudo-scenarios were used to compare the capabilities of three systems. A set of hypothetical problems was first created. Attempts were then made to address the problems using the different systems. The MERC system was included in this analysis to show the difference made by deliberate provisions for anonymity. The Elsewhere II (EW2) system was also included because the author believed it would represent most of the well-known social-oriented conversation systems on the Internet at the time.

A working version of each system was required for the analysis. There was no problem getting MERC to compile and run. Although the EW2 source code was available, it could not readily compile on the host computer used by the author. That was not a great concern as there was an abundance of services based on EW2 on the Internet.[129]

Scenario 1: Anonymous user harassing other users

The following set of scenarios compares the options available to deal with problematic anonymous users.

Scenario 1.1: When harassed by an anonymous user, what can user A (a non-superuser) do besides leaving the system or reporting the problem to a superuser?

Options	*T2*	*MERC*	*EW2/FH*
'Mute' the remarks of the harasser (with the help of the system).	Move conversation into personal chamber. Lock chamber (`chamber lock`). Request unknown users in chamber to leave. Remove unknown users by force if necessary (`chamber remove <user>`). Engage `earmuffs`.	-	`room bolt` `boot <user>` `earmuffs`
'Hide' from the harasser (by assuming a new identity).	`Morph` to a new username.	`Quit` and reconnect with new username.	`Quit` and reconnect with new username.

[129] The Foothills (FH) service was used in this analysis.

There were two distinct courses of action. Both options are supported by T2. However, not every system is designed to offer the first option.

Scenario 1.2: What can a superuser do when the harasser ignores every warning?[130]

Options	*T2*	*MERC*	*EW2/FH*
Stop harasser from making further remarks.	`freeze <user>`	`freeze <user>`	`freeze <user>`
Disable harasser's account (ie prevent harasser from using his/her username).	`deny <user>`	`deny <user>`	`banish <user>`
Disconnect harasser and reject future connections from harasser's computer.	Impose a thirty-day IP address ban on harasser (`autoban <user>`).[131]	Determine harasser's IP address (`users`). Impose an indefinite IP address ban on harasser (`ban <IP>`).	-
Impose 'virtual' fines.	-	Game punishment (eg deduction of gold, points, levels etc).[132]	-

Most systems allow some disciplinary action to be taken against a user without jeopardising the user's anonymity. The actions possible varies from system to system. MERC for example, requires a wizard (ie superuser) to know the network address of a user before connections from the user's computer can be banned. T2 and EW2 do not.

[130] What if the harasser were to use the tag technique to harass people? A senior T2 superuser could temporarily disable the tag technique (`tag off`).

[131] T2 records the harasser's IP address (in unencrypted form) in the *banned-IP* file.

[132] This will require the harasser's file to be editing from the operating system.

Scenario 1.3: The harasser reconnects from a different computer (ie different IP address or site), creates a new identity (ie new username), and continues harassment. What can a superuser do besides repeating the actions in Scenario 1.2?

Options	*T2*	*MERC*	*EW2/FH*
Temporarily prevent people from creating new username.	Temporarily close the system to visitors[133] (`vislock`).	Temporarily close the system to non-superusers (`wizlock`).	Temporarily close the system to visitors (`newbies`).
Reject all connection attempts from harasser's site.	The Head Superuser examines the banned-IP file to obtain harasser's IP address. Then banish harasser's site indefinitely (`ban <site>`).	Determine the harasser's IP address (`users`) and banish harasser's site indefinitely (`ban <site>`). E-mail a complaint to harasser's ISP.	Impose a ten minute site-ban on harasser (`splat <user> <minutes>`).
Report the problem to harasser's Internet Service Provider (ISP).	The Head Superuser examines banned-IP to obtain harasser's IP address (and therefore, ISP). Email complaint to ISP.	Checks the login logs (or harasser's player file) to determine harasser's IP address.	Determine harasser's IP address (`check ip` or `check info`).

Scenario 2: User realises own idiosyncrasies

Idiosyncrasies are a major hindrance to both authorship and identity anonymity. The following set of scenarios compares what a user with 'strong' idiosyncrasies has to do to attain conversational anonymity.

Scenario 2.1: User A realises that 'he' has a distinct style of writing. User A wants to be anonymous and make anonymous remarks. He chooses an anonymous username. What else should user A do?

Options	*T2*	*MERC*	*EW2/FH*
Keep sentences brief—eg 'yes' or 'no'.	Keep sentences brief.	Keep sentences brief.	Keep sentences brief.

[133] ie people without an existing account.

Options	*T2*	*MERC*	*EW2/FH*
Check for known idiosyncrasies in remarks before they are 'published'.	Use the word-blocker to check for idiosyncrasies.	Manual check.	Manual check.
Resort to authorship anonymity—people should not know whose idiosyncrasies they have observed.	Use the tag commands.	-	Use the `echo` command.
Use the pretend technique (ie a false name) and introduce engineered idiosyncrasies to confuse others.	Pretend technique.	Pretend technique.	Pretend technique.
Use a messenger to convey one's remarks.	Messenger.	Messenger.	Messenger.

While the task of overcoming problems with idiosyncrasies does not require help from the underlying conversation system, special provisions (eg the word-blocker in T2) can make the task easier.

Scenario 2.2: User A made a slip-up that caused his username to be non-anonymous. What can user A do in addition to changing his username?

T2	*MERC*	*EW2/FH*
Prevent repeating the slip-up. Set up the word-blocker to identify the slip-up.	-	-

The word-blocker can alert user A before he makes an identical slip-up.

Scenario 2.3: What if user A were not able to use a different username?

T2	*MERC*	*EW2/FH*
Conceal one's presence. Go into a permanent-Mirage (`mirage`). Converse from it.	-	-

Although, MERC provides a way for users to become 'invisible', it also provides a way for users to detect those that are invisible. In comparison, there is no simple way of checking who is hiding in a T2 permanent-Mirage.

Scenario 3: Introducing Anonymity into formal meetings

Scenario 3.1: What can a chairperson do to force every participant to adopt anonymity?

T2	*MERC*	*EW2/FH*
Anonymise every participant's remark Hold meeting in a Mirage (private-Mirage or permanent-Mirage).	-	-

Scenario 3.2:How would 'anonymous' and identified remarks be made?

Options	*T2*	*MERC*	*EW2/FH*
Switch between identified and anonymous usernames.	`Morph` between usernames. Everyone must do this together if meeting is not held in a private-Mirage.	Everyone must `quit` the system together and reconnect with new username.	Everyone must `quit` the system together and reconnect with new username.
Participants are asked to use their identified username but make use of provisions for making anonymous remarks.	Use the tag commands when anonymity is required. If meeting in a chamber, the owner can turn the chamber into a private-Mirage (`chamber mirage`) when anonymity is required.	-	Use the `echo` command when anonymity is required.

Scenario 3.3:How would a person respond to an anonymous remark in private?

T2	*MERC*	*EW2/FH*
A user can respond using the `tell <number> <remark>` command (or the `atell <number> <remark>` command if the user wanted to remain anonymous as well).	-	-

Scenario 3.4:How would the meeting be restricted to certain people?

T2	*MERC*	*EW2/FH*
Arrange to meet at chairperson's chamber.	-	Arrange to meet at chairperson's room.
Just before meeting commences, room is locked (`chamber lock`). Chairperson then removes unidentified users (`chamber remove <user>`).	-	`room bolt`. `room boot <user>`.
Use the tag commands to make anonymous remarks.	-	Use the `echo` command to make anonymous remarks.
Alternatively, chairperson turns the chamber into a private-Mirage when anonymity is required (`chamber mirage`).	-	-

Conclusion of the pseudo-scenario analysis

In Scenarios 2.2, 2.3, 3.1, and 3.3, T2 was able to provide a course of action when MERC and EW2 could not. In Scenario 3.4, T2 was able to provide solutions when MERC could not. In very simple terms, T2 had a four-scenario 'advantage' over the other systems. When T2 is compared to its former self (ie MERC), there was a five-scenario 'advantage'.

The author believed that these figures suggested that T2 was able to address more anonymity-related problems than MERC (and EW2). The author believed that T2 could provide better protection against loss of anonymity and better control over the use of anonymity.

5.3 AAC1 range analysis

The *AAC1 range analysis* involves calculating the AAC1 percentages for two scenarios—one representing a pro-anonymity (or high-anonymity) scenario while the other an anti-anonymity (or low-anonymity) scenario. The average between the two can then be used to represent the degree of anonymity supported by a system or environment.

Table 5.5 describes the result of an AAC1 range analysis on T2. The high-anonymity scenario could be seen as representing an infrequent anonymous user making ashouts from a permanent-Mirage. The low-anonymity scenario could be seen as representing a group of 'friends' conversing from within a locked chamber. T2 produced percentages that ranged from 13-50%. The average of 31% was used as the figure to represent T2.

Table 5.5: AAC1 range analysis of T2

	Factors	*High-Anon*	*Low-Anon*
1	Username concealed in high-anonymity scenario. Password-protected username used in low-anonymity scenario.	0/2	2/2
2	Personal details submitted? No.	0/2	0/2
3	Administration has access to personal details? No.	0/2	0/2
4	Compromised privacy? No.	0/2	0/2
5	Lack of Confidentiality? No.	0/2	0/2
6	Remarks tagged by username? Yes, in high-anonymity scenario.	0/1	1/1
7	Deanonymisation of anonymous remarks? No.	0/1	0/1
8	Known Idiosyncrasies? Users in the low-anonymity scenario know one another very well.	0/1	1/1
9	Private communication supported? Yes.	1/1	1/1
10	Meetings can be restricted by locking chamber. Closed meeting? Yes in low-anonymity scenario.	1/2	2/2
11	Proximity?	N/A	N/A
12	User is not anonymous to some participants. Yes in low-anonymity scenario.	0/2	2/2
	AAC1 Rating	2/18	9/18
	AAC1 Percentage	13%	50%
	Average Percentage	31%	

How much does T2 differ from Oz? The range analysis shows a 13% (44-31%) improvement in the level of protection—see Table 5.6 (on the next page).

Table 5.6: AAC1 range analysis of Oz

	Factors	*High-Anon*	*Low-Anon*
1	Username concealed in high-anonymity scenario. Password-protected username used in low-anonymity scenario.	0/2	2/2
2	Personal details submitted? No.	0/2	0/2
3	Admin has access to personal details? Only the head superuser can check user's IP address.	0.5/2	0.5/2
4	Compromised privacy? Conversations can be eavesdropped by (invisible) users and superusers.	2/2	2/2
5	Lack of Confidentiality? No.	0/2	0/2
6	Remarks tagged by username? Yes, in high-anonymity scenario.	0/1	1/1
7	Deanonymisation of anonymous remarks? No.	0/1	0/1
8	Known Idiosyncrasies? Users in the low-anonymity scenario know one another very well.	0/1	1/1
9	Private communication supported? Yes.	1/1	1/1
10	Meetings can be restricted by locking chamber. Closed meeting? Yes in low-anonymity scenario.	1/2	2/2
11	Proximity?	N/A	N/A
12	User is not anonymous to some participants. Yes in low-anonymity scenario.	0/2	2/2
	AAC1 Rating	4.5/18	11.5/18
	AAC1 Percentage	25%	64%
	Average Percentage	44%	

The range analysis was repeated for MERC and EW2—see Table 5.7 (below) and Table 5.8 (on the next page).

Table 5.7: AAC1 range analysis of MERC (continued next page)

	Factors	*High-Anon*	*Low-Anon*
1	Password-protected username is used in low-anonymity scenario.	1/2	2/2
2	Personal details submitted? No.	0/2	0/2
3	Admin has access to personal details? Several ranks of superusers can check user's IP address.	1/2	1/2
4	Compromised privacy? Conversations can be eavesdropped by (invisible) users and superusers.	2/2	2/2

	Factors	*High-Anon*	*Low-Anon*
5	Lack of Confidentiality? No.	0/2	0/2
6	Remarks tagged by username?	0.5/1[134]	1/1
7	Deanonymisation of anonymous remarks?	1/1[135]	N/A
8	Known Idiosyncrasies?	0/1	1/1
9	Private communication supported?	1/1	1/1
10	Meetings can be restricted? Closed meeting?	0/2	0/2
11	Proximity?	N/A	N/A
12	Users are not anonymous to some participants?	0/2	2/2
	AAC1 Rating	6.5/18	10/17
	AAC1 Percentage	36%	59%
	Average Percentage	47%	

Table 5.8: AAC1 range analysis of EW2/Foothills (continued next page)

	Factors	*High-Anon*	*Low-Anon*
1	Password-protected username is used in low-anonymity scenario.	1/2	2/2
2	Personal details submitted? To protect one's username, one has to supply one's e-mail address.	0/2	2/2[136]
3	Admin has access to personal details? Several ranks of superusers can check user's IP address.	1/2	2/2[137]
4	Compromised privacy?	0/2	0/2
5	Lack of Confidentiality?	0/2	0/2
6	Remarks tagged by username?	0/1[138]	1/1
7	Deanonymisation of anonymous remarks?	1/1[139]	N/A
8	Known Idiosyncrasies?	0/1	1/1

[134] If a user were able to be 'invisible', anonymous remarks would be possible.

[135] A user with the ability to 'detect invisibility' will be able to identify the remarks made by an 'invisible' user.

[136] To own a password-protected username, users have to disclose their e-mail address.

[137] Since user has provided an e-mail address, it can now be checked as well.

[138] The `echo` command can be used to remove username tags.

[139] Superusers are able to deanonymise echoes.

	Factors	*High-Anon*	*Low-Anon*
9	Private communication supported?	1/1	1/1
10	Meetings can be restricted? Closed meeting?	1/2	2/2
11	Proximity?	N/A	N/A
12	Users are not anonymous to some participants?	0/2	2/2
	AAC1 Rating	5/18	13/18
	AAC1 Percentage	28%	72%
	Average Percentage	50%	

The other well-known conversation system on the Internet is the Internet Relay Chat (IRC). Table 5.9 describes how IRC scored an average of 47%.

Table 5.9: AAC1 range analysis of IRC

	Factors	*High-Anon*	*Low-Anon*
1	Password-protected usernames are not supported.	0/2	0/2
2	Personal details submitted?	0/2	0/2
3	Admin has access to personal details? All superusers (or 'operators') know user's IP address.	1/2	1/2
4	Compromised privacy?	0/2	0/2
5	Lack of Confidentiality? User's IP address is publicised.	2/2	2/2
6	Remarks tagged by username? Yes, because anonymous remarks are not supported.	1/1	1/1
7	Deanonymisation of anonymous remarks?	N/A	N/A
8	Known Idiosyncrasies?	0/1	1/1
9	Private communication supported?	1/1	1/1
10	Meetings can be restricted? Closed meeting?	1/2	2/2
11	Proximity?	N/A	N/A
12	Users are not anonymous to some participants?	0/2	2/2
	AAC1 Rating	6/17	10/17
	AAC1 Percentage	35%	59%
	Average Percentage	47%	

Town Meeting (TM) was the final system analysed. The preliminary investigations concluded that TM was the most ideal environment for conversational anonymity. Is T2 more ideal than TM? Table 5.10 (on the next page) contains the results of TM.

Table 5.10: AAC1 range analysis of TM

	Factors	*High-Anon*	*Low-Anon*
1	Password-protected usernames are not supported.	0/2	0/2
2	Personal details submitted?	0/2	0/2
3	Admin has access to personal details?	0/2	0/2
4	Compromised privacy? The TM server logs all non-private remarks.	1/2	1/2
5	Lack of Confidentiality?	0/2	0/2
6	Remarks tagged by username? TM supports the nameless technique.	0/1	1/1
7	Deanonymisation of anonymous remarks? A flaw in TM may allow people to deanonymise anonymous usernames (and remarks).[140]	1/1	N/A
8	Known Idiosyncrasies?	0/1	1/1
9	Private communication supported?	1/1	1/1
10	Meetings can be restricted? Yes, a common-access password can be set. Closed meeting?	1/2	2/2
11	Proximity?	1/2	1/2
12	Users are not anonymous to some participants?	0/2	2/2
	AAC1 Rating	5/20	9/19
	AAC1 Percentage	25%	47%
	Average Percentage	36%	

[140] See page 16 of Ch 2.

Conclusions of the range analysis

Table 5.11: Summary of the AAC1 range percentages

Systems	*Average*	*Minimum*
T2	31%	13%
TM	36%	25%
Oz	44%	25%
IRC	47%	35%
MERC	47%	36%
EW2	50%	28%

As the author rightly concluded, TM was previously the 'best' environment for conversational anonymity. That 'title' should now belong to T2. T2 scored the lowest AAC1 range percentage—suggesting that it *protects anonymity better than TM, IRC, MERC, EW2, and Oz.* T2 also scored the lowest AAC1 range minimum (ie 13%)—suggesting that it is able to *provide the highest degree of anonymity.* Both scores suggest that T2 is the superior environment.

Chart 5.1 visualises the data in Table 5.11 along with the overall average and the standard deviation.

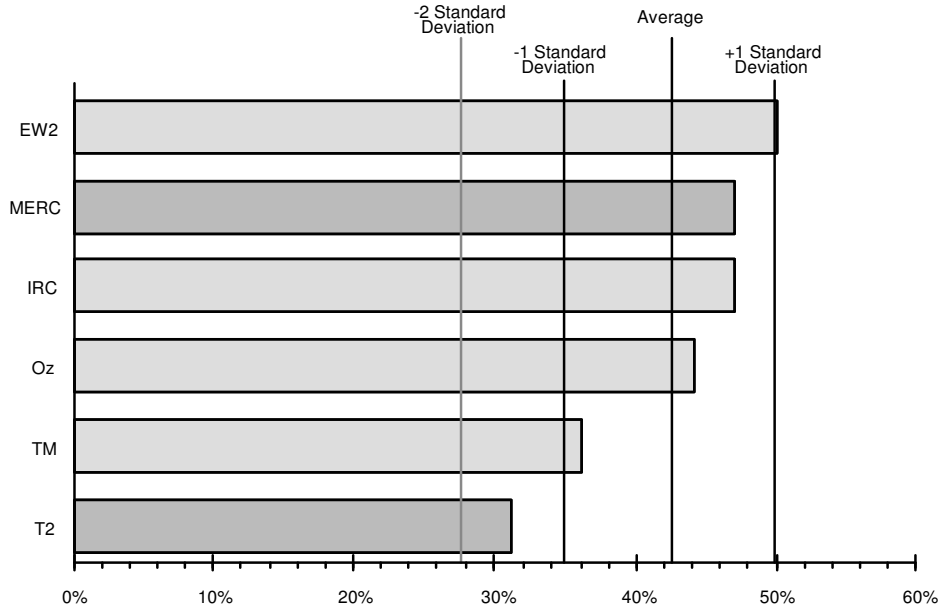

Chart 5.1: Comparing the potential for loss of anonymity

Only T2 and TM have a *below-average potential for loss of anonymity*. In other words, they were *above-average environments for conversational anonymity*. Except for T2, other environments laid within one standard deviation—

perhaps suggesting that T2 is a *distinctly* different (or rather, distinctly 'better') environment from the rest.[141]

5.4 Close of the in-depth research phase

The three goals of the in-depth research phase were to:

1 create a better method of introducing authorship anonymity into online conversations than the nameless or the nondescript-avatar technique,

2 create the ideal environment for conversational anonymity, and

3 create a set of guidelines and standards for supporting conversational anonymity.

Has the first goal been achieved? 'Yes,' because the tag technique is more ideal than the nameless technique. The tag technique does not hinder communication as the nameless technique does. Furthermore, it does not change the style of conversing as the nondescript-avatar technique did.

Has the second goal been achieved? T2 is a very ideal environment for conversational anonymity because anonymity is acknowledged, easily attained, protected, regulated, and allowed to co-exist with non-anonymity. The pseudo-scenario analysis and the AAC1 range analysis also showed that T2 is currently the best environment for conversational anonymity. The author believes that the second goal has been attained. Nevertheless, he hesitates to see T2 as the ultimate environment for conversational anonymity. Instead, he sees T2 as representing the *minimum standard of support*.

Has the third goal been achieved? 'No,' not yet.

A framework of guidelines and standards

The provisions (ie the policies and techniques) that had transformed MERC into T2 were therefore generalised and turned into a set of recommendations (and requirements). This compilation, dubbed the *Phantom Framework*, is found in Appendix 1 (on page 112).

[141] It is important that one does not think T2 has only been compared to TM, MERC, IRC, and EW2. Most conversation systems (on the Internet) are either based on or closely resemble these systems.

The Phantom Framework turns the outcomes of this research into a form that is convenient to disseminate, open to scrutiny, and independent of any specific method of implementation. This framework can now be used to raise the level of support (for conversational anonymity) of any conversation service to (and beyond) the benchmark set by T2.

The third goal, which is also the primary goal of this research, has now been achieved.

Chapter 6

Conclusion

Anonymity has been seen as a natural by-product of text-based online communication. The absence of face-to-face contact and the use of typed messages are sufficient to create anonymity. It is why a certain degree of anonymity is possible on a text-based conversation system that does not even claim to support anonymity.

This study has found that the naturally occurring anonymity is not ideal. A better 'kind' of anonymity has been found—one that is more practical and secure—one that has been deliberately created. Such a kind of anonymity was found in T2 and T1 (and to a lesser extent in Oz). Such a kind of anonymity can be recreated by the complete adoption of the *Phantom Framework*.[142]

Apart from the Phantom Framework, this study has also created a body of knowledge. Concepts such as *absolute-anonymity, profiled-anonymity, confined-anonymity, hidden-anonymity, protected-anonymity, conversational anonymity, identity anonymity,* and *authorship anonymity* will now be available for others to use. The *Anti-Anonymity Checklist 1 (AAC1)* analyses and the *PANIC notation* should also be available.

This work has also shown that anonymity does not need to be a hindrance to identity (ie non-anonymity), and vice-versa. Both can coexist and complement one another. The benefits of both can be attained at the same time. Support for the tag technique and password-protected usernames should enable a group of people to be identified, and yet converse anonymously.

Although the goals of this research have been met, the author believes that there is room for further research. The author has responded to the problem of idiosyncrasies by supporting keyword screening and a de-capitalisation mechanism. The idea of a style scrambler capable of removing one's idiosyncrasies or adding foreign idiosyncrasies was not implemented. Had Flinn and Maurer's style scramblers been possible, the degree of a user's

[142] See Appendix 1 (page 112).

anonymity would have been elevated to new heights. Implementing Flinn and Maurer's style scramblers is an area for further research. Another possible area is to expand the Phantom Framework to support other online activities such as e-commerce[143] and collaborative writing (or work).[144] Further work is also needed before the Phantom Framework can be made acceptable to standards-setting bodies in particular, the *International Standards Organisation*.

The conclusion or thesis of this research is simple—*anonymity, if desired, should be deliberately supported*. Should every online service support conversational anonymity? No, but at least one should. T1 was that service for a while. It has now been closed. Perhaps until T2 opens or until the Phantom Framework is adopted by one of the many conversation services on the Internet, people needing anonymity may have to face the kinds of difficulties they have been always had to face.

[143] See Appendix 2 for a hint of what might be possible.

[144] See Lee 94 for a hint of what might be possible.

Appendices

Appendix 1: The Phantom Framework

The Phantom Framework does not contain of specifications but rather a combination of restrictions, demands, and examples. It specifies what the ideal environment for conversational anonymity is but does not specify how it should be created. The Phantom Framework is intended to help system designers and administrators transform an existing conversation service into a more adequate environment for users that require anonymity.

Each clause is permanently referenced using a <section>/<clause> (<subclause>)-<release> scheme—eg [12/1(1)-1]. Alpha characters are used to relate certain clauses together—ie [12/1/(1*a*)-1] and [12/1/(1*b*)-1]. All future amendments will have to be introduced as additional clauses with a new release number. An existing clause cannot be modified after official release.

The Phantom Framework 1 (PF-1)

0/	Notes
0/1(n)-n	*Release history—a record of versions (where n should become the release version), authors, dates, release channel, and comments.*
0/1(1)-1	*Phantom Framework 1 (PF-1)* by *Andrew Lee* Published on 7th September 2000 at `http://www.geocities.com/drew_drew.geo/phantom/`
0/2a-1	The Phantom Framework (PF) clauses are consistently qualified by a scheme of reserved words. A clause qualified by 'must' means a mandatory requirement.
0/2b-1	A clause qualified by 'shall' means a conditional requirement, where adherence is mandatory if the execution is possible or applicable.

0/2c-1	A clause qualified by 'should' means an adaptable requirement, where the terms or level of adherence can be changed.
0/2d-1	A clause qualified by 'may' means a suggestion or discretionary requirement.
0/3-1	To claim compliance to the Phantom Framework, the *musts* and *shalls* in sections 1–11 **must** be implemented or observed.
0/4-1	A 'service' (as opposed to 'services') **shall** consist of the system (providing the services), services (ie the aid provided), policies (ie the terms, conditions, and codes of practice), and the administration (ie the people that maintain the service).
0/5-1	A 'system' **shall** consist of programs and the standard procedures that revolve around the programs.
0/6-1	The 'user' is a person using the service. A user may be equated to a customer.
0/7-1	An 'administrator' means a member of the administration. It **shall** not imply any rank or responsibilities.
0/8-1	A 'superuser' is a user who is given special authorities and privileges by the administration to provide services to other users. A superuser is a member of the administration but the reverse may not be true.
0/9-1	The administration **must** have complete control over the system and data (or official records). The programs **must** be understood and modifiable.
0/10-1	Every member of the administration **must** operate under a consistent code of practice. No administrator **shall** be exempted from the code of practice.
0/11-1	The administration **must** be able to give and honour guarantees to a user.
0/12-1	A 'contribution', 'remark', or 'message' **shall** imply a text-only content.
0/13-1	A piece of 'real-life user information' **shall** mean a truth or fact about a person. A person's full real name, date of birth, employer, telephone number, or e-mail address are examples of real-life user information. A user's username may or may not be treated as a real-life user information.
0/14-1	Real-life user information is gathered 'indirectly' when it is obtained without the explicit knowledge of the user. Potential sources may include a user's personal and formal affiliations, public databases, the user's computer or software (eg the operating system or Telnet client), or the network protocol.

1/ The Limited Information Policy

1/1-1	A service **must** be able to function without knowing its users' real-life identity. A service should function without requiring any piece of real-life user information.

1/2-1	Real-life user information **shall** not be gathered (whether directly or indirectly) until a user has rejected anonymity or violated his or her rights to anonymity (ie violated the *contractual-anonymity understanding* [5/1-1]).
1/3-1	Any unavoidable need, possession, or use of a piece of real-life user information **must** be shown (ie proved) to be in the direct interest of the user or community (ie other users). Nevertheless, a justification **shall** not automatically imply an approval to gather, store, or use a piece of real-life user information.
1/4-1	A user **must** be notified before (and not after) any indirect gathering [0/14-1] of real-life user information, even if the service is not seeking the user's permission.

2/ The Confidentiality Policy

2/1-1	Users **must** be allowed to hide their real-life identity from other users. By default, every piece of real-life user information **must** be presumed confidential and guarded from the public (ie other users). The user **shall** decide what is not confidential.
2/2-1	Justifications [1/3-1] and *usage-access details* (UAD) **must** be given to a user before a piece of real-life user information is stored (ie permanently recorded). The UAD **must** describe who will have access to what piece of user real-life user information explicitly.
2/3-1	The use and disclosure of any real-life user information **must** be in accordance with the UAD [2/2-1].
2/4-1	Any record of a user's real-life identity should be kept in an encrypted form and should be deleted when it has served its purpose.
2/5-1	A user's authorisation (ie consent) **must** not be implied or extended. A user who authorises a specific member of the administration does not automatically authorise the entire administration.
2/6-1	The UAD **must** be comprehensive at the onset. The administration **shall** not modify the UAD without (formally) notifying a user a few days (not hours) in advance so that the user may be able to take any necessary action.
2/7-1	The UAD **must** never be knowingly violated.
2/8-1	An administrator who has access to any piece of real-life user information **must** be subjected to additional identification and authentication than the usual level imposed upon a user. Identification may proceed beyond password identification and require real-life names and e-mail addresses to be disclosed. Authentication may become more subjective and involve interviews to verify an administrator's identity.

3/ The Transparency Policy

3/1-1	Users **must** know what information is kept about them. The administration **must** guarantee that every piece of real-life user information in possession has been disclosed.

3/2-1	Any information generated by the administration that is not considered a piece of real-life user information [0/13-1] (eg comments about a user's 'misbehaviour') **shall** remain the property of the service and **shall** not have to be revealed to a user.

4/ The Privacy Policy

4/1-1	Activities of non-superusers (ie users) **must** never be secretly monitored (ie observed or recorded). Users **must** be aware of any monitoring in advance and while it occurs.
4/2-1	Any unavoidable monitoring **must** be shown (ie proved) to be in the direct interest of the user or community.
4/3-1	A group of users **shall** be able to hold a private meeting. No user (including a superuser) **shall** be able to join (ie intrude) the private meeting. *Private lockable chambers (or channels)* should be implemented.

5/ The Contractual-Anonymity Policy

5/1-1	An explicit (written) understanding **must** exist between the administration and the users whereby anonymity is guaranteed provided a user observes certain conditions. The *contractual-anonymity understanding* (CAU) **shall** specify the conditions, guarantees, and repercussions. The repercussions may include efforts to determine a violator's real-life identity.
5/2-1	A person **must** be presented with the CAU before the person becomes a user (ie before the person uses the service).
5/3-1	Anonymity may be granted to a user that has not explicitly agreed to the conditions in the CAU. The repercussions (outlined the CAU) may still be enforceable if the conditions are violated.
5/4-1	A user **must** be shown (ie proven) to have explicitly violated the CAU before any repercussions are enforced. The service **must** not knowingly jeopardise the anonymity of any user while verifying a user's compliance (to the CAU). In fact, a user should not be investigated unless there is due cause (ie evidence).
5/5-1	A repercussion that is not described or implied in the CAU **shall** not be considered (or authorised) if it will jeopardise the anonymity of the violator. The service **must** not knowingly violate the CAU.
5/6-1	The conditions and repercussions in the CAU should be proportional to the (benefits of the) provisions (in the CAU). Support for higher degrees of anonymity should attract more serious repercussions.
5/7-1	Any guarantee (relating to anonymity) that a user cannot verify or the administration cannot prove **shall** be put in writing (in the CAU or elsewhere).
5/7-1	The CAU **must** be comprehensive at the onset. The administration **shall** not modify the CAU without (formally) notifying the users a few days in advance.

6/	Anonymity-Compatible Operations *An anonymity-compatible method of operation does not endanger or reduce the anonymity of the users.*
6/1-1	A standard procedure or authorised practice **must** not violate the CAU [5/1-1].
6/2a-1	A standard procedure **shall** not burden an administrator or superuser with secrecy. A standard procedure that will expose an administrator or superuser to a piece of real-life user information should be automated.
6/2b-1	The ban-by-username technique [12/9] should be implemented.
6/3-1	Standard procedures that may jeopardise the anonymity of users **must** be identified and monitored. Any potential of jeopardy anonymity **must** be remove if monitoring were not possible.
6/4-1	An updated list of the administrative team along with their ranks, responsibilities, and capabilities **shall** be available to the users. However, it is not necessary to reveal the real-life identity of an administrator.
6/5-1	A superuser **must** be identified in a manner that cannot be forged by a non-superuser. A user **must** be able to verify that a particular user is indeed a superuser. A user should be able to list all the superusers on duty.
6/6-1	The administration **shall** address a user by the user's chosen username in any conversation or documentation. An anonymous alias **shall** be assigned to a person if one does not exist (or is not appropriate).

7/	The User-Discretion Policy (Support for Identification)
7/1-1	A user **shall** be allowed be non-anonymous (ie possess an identified username and make identified remarks). Some form of user identification and authentication *(eg password-protected usernames)* **must** be supported. A user **shall** be able to make remarks that are tagged by his or her username.
7/2-1	Real-life user information **shall** not be used for the identification and authentication of a (typical) user. Identification and authentication **shall** not be an acceptable justification for requiring or gathering any piece of real-life user information. *A correct password should be sufficient.*
7/3-1	Where there is a conflict between anonymity and identification, anonymity **shall** have precedence. Provisions for identification may be revoked to preserve anonymity. *A Mirage [12/6] for example, forces everyone to be anonymous for there to be sufficient decoys for authorship anonymity [8/].*

8/	**Support for Authorship Anonymity** *Authorship Anonymity is created when a user is able to make a remark that cannot be traced to the user.*
8/1-1	The system **shall** support at least one authorship anonymity technique. The *tag technique* [12/1] should be implemented.
8/2-1	The system **must** enable the source of a contribution to be concealed from users and the administration. The system **must** not provide other users or the administration a way to prove the source of a contribution (ie it **must** not technically be possible to deanonymise an anonymous contribution).
8/3-1	An authorship anonymity technique uses a crowd to conceal the contributor and disperse any implications. The theoretical minimum should be the presence of four candidates including the actual contributor (or three decoys). A 'candidate' is a user who could have made the contribution technically. An authorship anonymity technique **shall** be disabled if the minimum is not satisfied. Each candidate should be a different person. A policy **shall** be imposed to prohibit multiple online presence (ie *multiplaying*).
8/4-1	The system should provide users with the ability to reject (ie not receive) anonymous remarks.
8/5-1	The system **must** provide the administration with the ability to disable all authorship anonymity techniques (hence, stop the creation of anonymous remarks).

9/	**Support for Identity Anonymity** *Identity Anonymity is created when a user's real-life identity (ie full real name) is not known or cannot be verified.*
9/1-1	The system **shall** support at least one identity anonymity technique. The *alias on-the-fly technique* [12/4] should be implemented.
9/2-1	The system should not collect, force the disclosure, or publicise any information that is directly associated to a user's real-life identity *(eg a user's e-mail address or IP address)*.
9/3-1	The system should provide users with the ability to isolate themselves from anonymous users.
9/4-1	The system **must** provide the administration with the ability to stop the creation (or use) of anonymous usernames.

10/	**Anonymity Guards** *Anonymity guards help prevent avoidable loss of anonymity.*
10/1-1	A user **must** show a need for anonymity before an anonymity guard may interfere with the actions of the user. For example, the activation of the anonymous mode [12/5] or being in a Mirage [12/6] is sufficient evidence that a user requires anonymity.
10/2-1	The contents of a user's contribution (ie message) should not be changed without the user's knowledge. Some form of explicit approval may be required before the system makes any changes.

11/ Awareness (Acknowledgement) & Education

11/1-1	Support for (conversational) anonymity **must** be declared and **shall** be publicised. The keyword *anonymity* or *anonymous* **shall** be used. All endorsements and compliance to standards **shall** be declared (and elaborated) and **shall** be publicised.
11/2-1	The administration **must** express in writing, all the asserted guarantees (ie assurances) and provisions for anonymity that a user may not know about or be able to verify.
11/3-1	Users **must** be able to examine the administrative code of practice. In fact, users should be presented with this code before using the service.
11/4-1	Users should be advised how to use the provisions and the risks (if any) associated to each provision.
11/5-1	The mechanics behind every (anonymity) provision should be documented and made available to users. This may be important for user confidence and endorsement purposes.

12/ Supplementary Clauses

12/1 The Tag Technique

12/1(1)-1	The tag-based communication commands (ie tag commands) **shall** resemble their untagged (ie non-anonymous) counterparts in form (ie syntax, feedback, and presentation) and function. However, usernames **must** be removed and replaced by numbers.
12/1(1a)-1	An anonymous say command **shall** be implemented. `Asay` **shall** enable a user to speak anonymously at a gathering (ie to the people in the same room or channel as the user). Syntax: `asay <message>` Feedback (to the user): `[<number> you] say '<message>'.` Presentation (ie output): `[<number>] says '<message>'.`
12/1(1b)-1	Two anonymous tell commands **shall** be implemented. The first form of `atell` **shall** enable users to send private anonymous message to specific users. Syntax: `atell <user> <message>` Feedback: `[<num> you] tell <user> '<message>'.` Presentation: `[<num>] tells you '<message>'.` The second form of `atell` **shall** enable users to reply a tagged remark anonymously and privately. Non-delivery of an atell <number> command **shall** not be reported to the user. Syntax: `atell <num> <message>` Feedback: `[<num> you] tell [<num>] '<message>'.` Presentation: `[<num>] tells [<num> you] '<message>'.`

12/1(1c)-1 An anonymous shout command **shall** be implemented. `Ashout` **shall** enable users to make anonymous remarks to everyone online.

Syntax: `ashout <message>`
Feedback: `[<num> you] shout '<message>'.`
Presentation: `[<num>] shouts '<message>'.`

12/1(1d)-1 An anonymous emote command **shall** be implemented. `Aemote` **shall** enable users to convey gestures anonymously at a gathering.

Syntax: `aemote <activity>`
Feedback: `[<num> you] <activity>.`
Presentation: `[<num>] <activity>.`

12/1(2)-1 The tag **shall** be a number incremented from the last number used. The default range of numbers **shall** be 1 to 20. At the end of the sequence, the initial number **shall** be reused. If the duration between the assignment of the first number (ie 1) and the last number (ie 20) were less than one minute, the maximum (ie the range) should be automatically increased. If the duration between the assignment of the first and last number were more than a minute, the default range (ie 1–20) should be used. If a number were not reassigned after ten minutes, it should be cleared (ie it should not refer to any user).

12/1(3)-1 Explicit evidence that connects a user, a tag, and a contribution **must** not be permanently stored or accessible to anyone including the administration—see [8/2-1].

12/1(4)-1 The anonymous mode [12/5], decoy-checker [12/2] and the capitalisation checker [12/3] **shall** be implemented as accessories to the tag technique.

12/1(5)-1 The anonymous rooms [12/6] may be implemented as accessories to the tag technique.

12/2 The Decoy Checker

12/2(1)-1 The decoy checker mechanism **shall** disable all authorship anonymity techniques (such as the tag technique [12/1]) when there are fewer than four active participants in the meeting.

12/2(2)-1 The decoy checker **must** discount inactive participants (ie users that could not have made any remark). It **must** rely on the information that the system makes available to the users when discounting candidates. *For example, if the system were to report broken network connections then a disconnected user **shall** not be counted.*

12/3 The Capitalisation Checker

12/3(1)-1 All capitalisation **must** be removed from an anonymous message. A user's consent **shall** not be necessary for such intervention.

12/4 The Alias on-the-fly Technique

12/4(1)-1 The system **shall** allow a user to change usernames without alerting other users.

12/4(2)-1	The `morph` command **shall** be implemented. Users **shall** be asked to confirm their decision to change usernames. The usual login (ie identification) procedure **must** proceed before a user is given a new username. Certain welcome or introductory messages may be omitted to expedite the change.

12/5 Anonymous Mode

12/5(1)-1	When a user activates the anonymous mode, all his or her contributions (ie remarks) **shall** be anonymised using an authorship anonymity technique (such as the tag technique [12/1]). The decoy-checker mechanism [12/2] **shall** be activated when the anonymous mode is active.
12/5(2)-1	The system **shall** stop a user from making any action that may be traced back to the user. For example, the system should stop the user from using social commands.
12/5(3)-1	A user **shall** be able to activate or deactivate the anonymous mode at any time.

12/6 Anonymous Rooms/Channels (Mirages)

12/6(1)-1	An authorship anonymity technique **shall** be imposed on every person in a Mirage (including superusers). The tag technique [12/1] should be implemented.
12/6(2)-1	The system **shall** not reveal the number of users in a Mirage. The decoy-checker mechanism [12/2] **shall** be activated in a Mirage but **shall** only prohibit the use of an authorship anonymity technique if the total number of candidate users on the system is fewer than the theoretical minimum of four.
12/6(3)-1	The system **must** suppress any system message that may expose the presence of a particular user in a Mirage. For example, the system **shall** stop the use of social commands and **shall** enable users to enter and leave a Mirage without generating the usual entry or exit notifications.
12/6(4)-1	The system **must** stop the execution of any command that may verify the presence of a particular user in a Mirage. For example, the system **shall** disable any command that reveals a local participant list (eg `where` and `look`) and any command that is applied to another local user (eg `look <user>`).
12/6(5)-1	Users **shall** be able to transform their private chambers [4/3-1] into private-Mirages. The system **must** notify everyone in the chamber (or channel) when miraging and de-miraging occurs.
12/6(6)-1	The elimination of global participant lists [12/7] may be implemented as an accessory to Mirages.

12/7 The Elimination of Global Participant lists

12/7(1)-1	All global participant lists **shall** be removed or replaced by statistics (eg a count of the number of users online).
12/7(2)-1	Local participant lists **shall** not be removed (as this may inhibit private non-anonymous conversations).

12/8 The Word-blocker

12/8(1)-1	The word-blocker **shall** alert the presence of any banned phrase or word from a message and await the user's decision whether to allow the message to be communicated.
12/8(2)-1	The list of banned words used by the word-blocker **shall** be considered sensitive and **must** only be accessible by the rightful user.
12/8(3)-1	The word-blocker should be implemented on a front-end (ie a user-side client software) and should be independent from the server (or service).

12/9 The Ban-by-Username Technique

12/9(1a)-1	An `autoban <user>` command **shall** enable a superuser to ban a particular IP address by specifying the user's username. The system **shall** not reveal the user's IP address to the superuser. The system **shall** determine the IP address of a user (from the temporarily login [12/10(2)-1] or problem-user [12/10(3)-1] logs) or determine the IP address from the network protocol and impose a ban of the IP address.
12/9(1b)-1	Bans on IP addresses should be recorded in a *banned-IP* file. Each case should have a fixed expiry date. Upon expiry, the system should automatically remove the IP address ban. The banned-IP file does not need to be encrypted.
12/9(2)-1	An `unban <user>` command **shall** enable a superuser to remove the fixed-period IP address ban on a particular user.

12/10 Temporary IP address Logs

12/10(1)-1	Every user's IP address **shall** be logged by the system. This activity **must** be known to all users. The IP address will enable the service to impose a site ban or to contact a violator's Internet Service Provider. All repercussions **shall** be outlined in the CAU [5/1-1].
12/10(2)-1	The login details **shall** be stored in encrypted form [2/4-1] in a *temporary login log*. Access to the temporary login logs **shall** be monitored, justified, and confined [6/3-1]. A `getip <user>` command would satisfy the three conditions. It would only return the IP address of a specific user.
12/10(3)-1	An entry in the temporary login log **shall** have a fixed expiry period. The recommended period **shall** be twenty-four hours. An entry **shall** be deleted when it expires. A `saveip <user>` command **shall** allow the IP address of certain users to be retained (indefinitely) in a *problem-user log*. A `deleteip <user>` command should enable a user's IP address to be deleted from the problem-user log. The problem-user log does not need to be encrypted.

Appendix 2: Anonymity as a commercial service

The following scenario shows how the anonymity provisions (in T2) could be used to provide a commercial service.

'Dr Kelly Brown' wanted to interview people that were suffering from a very rare disease. She could not find any local patients to interview. She turned to the Internet. She found an anonymity provider on the Internet. She found the terms and fees acceptable. She registered and made a reservation. Her meeting details were recorded and approved. She was then issued a unique account number. The account number identified Dr Brown to the system.

The system calculated the fee and prompted her about the payment method. She chose to pay using a cheque. She was given the necessary information (ie the payee, the billable amount in her local currency, and due date) and instructions. The service did not know or needed to know her real-life identity. The cheque could be written by anyone as long as her account number was quoted.

Dr Brown mailed her cheque. A week later, Dr Brown logged on the web server (with her account number) to check if her payment had been received. Since it was, she was given further instructions.

Once Dr Brown understood what was involved, she began making announcements in medical journals, Internet newsgroups, and notice boards at the local hospitals. She described who she was, what she wanted to do, the benefits of the interviews, the hows and whens, and guarantees anonymity. She specifically mentioned the use of an independent professional service to provide anonymity.

Half an hour before the scheduled meeting, the system automatically created a virtual meeting room for Dr Brown. As instructed, Dr Brown arrived fifteen minutes before the meeting. She tried to use the username 'Kelly'. Unfortunately, that username had already been reserved. She chose 'Kel' instead. Upon entry into the system, she was presented with a list of active meetings. She found the meeting with the title that she had specified (when she made her reservations). She was then brought into the appropriate meeting room. She then protected her username using her unique account number (as she had been instructed in advance). The system recognised the account number and now recognised 'Kel' as Dr Brown. The system assigned her special privileges because she was the host. Even if Dr Brown were to change her character's password, the system would continue to recognise 'Kel' as Dr Brown (until 'Kel' was deleted).

The system began billing Dr Brown once the scheduled meeting time arrived. There were no visitors in the first hour. Dr Brown had purchased five meeting

hours, and so four remained. In the second hour, one person arrived. Dr Brown began her interview immediately. More people began arriving.

When the meeting was over, Dr Brown reminded everyone to reserve (ie password protect) their username for future meetings. Dr Brown had requested (and paid for) a facility to store the usernames of her guests. As long as her account was active, her guests would also be stored. Dr Brown had paid to keep her account active for a year. When all her guests had left, she closed the meeting. The remaining meeting time was credited to her account.

Dr Brown's guests were anonymous to her although she was not anonymous to them. She could have been anonymous if she had wanted to since the service would have protected her identity. She could have attained protected-anonymity from the service. Even if Dr Brown could recognise her guests at later meetings, they would still be anonymous. They would experience profiled-anonymity unless they had identified themselves to Dr Brown. Even if they did, they should still be able to make anonymous remarks (if they knew about the tag commands).

Cited Works

Bruckman, Amy. (1992). *Identity Workshop: Emergent Social and Psychological Phenomena in Text-Based Virtual Reality*. Massachusetts Institute of Technology Media Laboratory. ftp://ftp.cc.gatech.edu/pub/people/asb/papers/identity-workshop.rtf

Crumlish, Christian. (1997). *Chatting, Conferencing, and Virtual Worlds.* **The Internet for Busy People (2nd Edition)**. Osborne/McGraw-Hill: 221–244.

Detweiler, L. (1993a). *Anonymity FAQ*. Anonymous FTP to rtfm.mit.edu:/pub/usenet/news.answers/net-anonymity/

Detweiler, L. (1993b). *Privacy & Anonymity FAQ.* Anonymous FTP to rtfm.mit.edu:/pub/usenet/news.answers/net-privacy/

Ellis, C A. Gibbs, S J. & Rein, G L. (1991). *Groupware: Some Issues and Experiences*. **Communications of the ACM**, 34(1): 38–58.

Flinn, Bill. & Maurer, Hermann. (1995). *Levels of Anonymity*. **Journal of Universal Computer Science**. 1(1): 35–47.

Froomkin, Michael. (1995). *Anonymity and Its Enmities*, **Journal of Online Law**, World Wide Web, http://www.law.cornell.edu/jol/froomkin.htm

Lee, Andrew. (1993). *ViE: A Semi-Anonymous Interaction Environment*. Working Paper 1993-3-098/B. School of Information Technology, Bond University, Australia.

Lee, Andrew. (1994). *Anonymous Collaboration: An Alternative Technique for Working Together*. **SIGCHI Bulletin**. 26(3): 40–46.

May, Tim C. (1994). *Cyphernomicon*. World Wide Web, http://ocaxp1.cc.oberlin.edu/~brchkind/cyphernomicon/

Morningstar, Chuck. & Farmer, F Randy. (1992). *The lessons of Lucasfilm's Habitat*. In Benedikt (Ed). **Cyberspace: First Steps**. MIT Press.

Nunamaker, J F. Dennis, A R. Valacich, J S. Vogel, D R. & George, J F. (1991). *Electronic Meeting Systems To Support Group Work*. **Communications of the ACM**, 34(7): 40–61.

Rees, Michael J. Iannella, Renato. Lee, Andrew. Smith, Glen. & Woo, T K. (1993). *Spinning a Yarn: User Interfaces for Synchronous Remote Electronic Meetings*. In **Proceedings of OZCHI93**. OZCHI.

Reid, Elizabeth M. (1991). *Electropolis: Communication and Community on Internet Relay Chat*. Honours Thesis, University of Melbourne, Australia. ftp://parcftp.xerox.com/pub/MOO/papers/electropolis.txt

Reid, Elizabeth M. (1993). *Cultural Formations In Text-Based Virtual Realities*. Masters Thesis, University of Melbourne, Australia. ftp://parcftp.xerox.com/pub/MOO/papers/Cultural_Formations.txt

Rigby, Katrina. (1995). *Anonymity on the Internet must be Protected*. Research Paper 6.805/STS085, Massachusetts Institute of Technology. World Wide Web, http://swissnet.ai.mit.edu/6095/student-papers/fall95-papers/rigby-anonymity.html

Shafer, Kevin. (1997). **Novell's Dictionary of Networking**. Novell Press.

Smith, Jennifer. (1992). *MUD FAQ.* Anonymous FTP to rtfm.mit.edu:/pub/muds/misc/mud-faq

Suler, John. (1997a). *TextTalk: Psychological Dynamics of Online Synchronous Conversations in Text-Driven Chat Environments*. World Wide Web, http://www1.rider.edu/~suler/psycyber/texttalk.html

Suler, John. (1997a). *The Bad Boys of Cyberspace: Deviant Behavior in Online Multimedia Communities and Strategies for Managing it*. World Wide Web, http://www1.rider.edu/~suler/psycyber/badboys.html

Wallace, Patricia. (1999). **The Psychology of the Internet**. Cambridge University Press.